```
STMNT    LOC     OBJECT CODE      SOURCE STATEMENT

  1                               ;     COMPLIMENTS OF PROCESSOR TECHNOLOGY
  2                               ;
  3                               ;     A FOCAL INTERPRETER FOR THE 8080 MICROPROCESSOR
  4                               ;
  5                               ;  FOCAL IS A REGISTERED TRADEMARK OF DIGITAL EQUIPEMENT CORPORATION
  6                               ;
  7                               ;     WRITTEN BY ROBERT ARNSTEIN      FALL 1975
  8                               ;                5628 MEADOW CREST DRIVE
  9                               ;                DALLAS, TEXAS 75230     (214) 368-6820
 10                               ;
 11                               ;
 12     0000                      XXX:      EQU     0
 13     0001                      PROCT:    EQU     1          ;ASSEMBLE FOR PROCESSOR TECHNOLOGY
 14     0000                      INTEL:    EQU     0          ;ASSEMBLE FOR INTELLEC/80
 15     0000                      INTRP:    EQU     0
 16                                         IF      PROCT
 17     0000                               ORG     0
 18     0000     C3  64  04                JMP     START
 19     0003         00  20      SPAD:     DW      8192
 20                                         ENDF
 21                                         IF      INTEL
 22                                         ORG     10H
 23                               SPAD:     DW      1000H
 24                                         ENDF
 25                                         IF      XXX
 26                                         ORG     0
 27                                         JMP     START
 28                               SPAD:     DW      8192
 29                                         ENDF
 30     0005                      SBANK     EQU     $
 31                               MULX4:
 32     0005     C6  00                    ADI     0          ;ADD OPERAND 3RD FRACTION
 33     0006                      MULP3:    EQU     >$-1
 34     0007     5F                        MOV     E,A        ;4TH PARTIAL PRODUCT
 35     0008     7A                        MOV     A,D        ;3RD PARTIAL PRODUCT
 36     0009     CE  00                    ACI     0          ;ADD OPERAND 2ND FRACTION
 37     000A                      MULP2:    EQU     >$-1
 38     000B     57                        MOV     D,A        ;3RD PARTIAL PRODUCT
 39     000C     79                        MOV     A,C        2ND PARTIAL PRODUCT
 40     000D     CE  00                    ACI     0          ;ADD OPERAND 1ST FRACTION
 41     000E                      MULP1:    EQU     >$-1
 42     000F     C3  EB  11                JMP     MULX5      ;TO ROM CODE
 43                               ;     RAM DIVIDE SUBROUTINE
 44                               DIVX5:
 45     0012     D6  00                    SUI     0          ;SUB DIVISOR 4TH FRACTION
 46     0013                      OP4S:     EQU     >$-1
 47     0014     7D                        MOV     A,L        ;REMAINDER 3RD FRACTION
 48     0015     DE  00                    SBI     0          ;SUB DIVISOR 3RD FRACTION
 49     0016                      OP3S:     EQU     >$-1
 50     0017     6F                        MOV     L,A        ;REMAINDER 3RD FRACTION
 51     0018     7C                        MOV     A,H        ;REMAINDER 2ND FRACTION
 52     0019     DE  00                    SBI     0          ;SUB DIVISOR 2ND FRACTION
 53     001A                      OP2S:     EQU     >$-1
 54     001B     67                        MOV     H,A        ;REMAINDER 2ND FRACTION
 55     001C     7B                        MOV     A,E        ;REMAINDER 1ST FRACTION
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

  56     001D   DE  00                   SBI     0           ;SUB DIVISOR 1ST FRACTION
  57     001E                    OP1S:   EQU     >$-1
  58     001F   5F                       MOV     E,A         ;REMAINDER 1ST FRACTION
  59     0020   3E  00                    MVI     A,0         ;REMAINDER 4TH FRACTION
  60     0021                    OP4A:   EQU     >$-1
  61     0022   C9                       RET     ;RETURN TO ROM
  62                            DIVX6:
  63     0023   C6  00                    ADI     0           ;ADD DIVISOR 3RD FRACTION
  64     0024                    OP3A:   EQU     >$-1
  65     0025   6F                       MOV     L,A         ;REMAINDER 3RD FRACTION
  66     0026   7C                       MOV     A,H         ;REMAINDER 2ND FRACTION
  67     0027   CE  00                    ACI     0           ;ADD DIVISOR 2ND FRACTION
  68     0028                    OP2A:   EQU     >$-1
  69     0029   67                       MOV     H,A         ;REMAINDER 2ND FRACTION
  70     002A   7B                       MOV     A,E         ;REMAINDER 1ST FRACTION
  71     002B   CE  00                    ACI     0           ;ADD DIVISOR 1ST FRACTION
  72     002C                    OP1A:   EQU     >$-1
  73     002D   5F                       MOV     E,A         ;REMAINDER 1ST FRACTION
  74     002E   3E  00                    MVI     A,0         ;REMAINDER 4TH FRACTION
  75     002F                    OP4X:   EQU     >$-1
  76     0030   C3  57  12                JMP     DIVX2       ;TO ROM CODE
  77                            ;       RAM LOCATIONS USED BY THE BINARY
  78                            ;       FLOATING POINT SYSTEM
  79     0033                    OVER:   EQU     >$
  80     0033           00               DB      0           ;INITIALLY CLEAR
  81     0034                    PREX:   EQU     OVER+1      ;PREVIOUS EXPONENT
  82     0035                    ACCE:   EQU     PREX+1      ;ACCUMULATOR EXPONENT
  83     0036                    ACCS:   EQU     ACCE+1      ;ACCUMULATOR SIGN
  84     0037                    ACC1:   EQU     ACCS+1      ;ACCUMULATOR 1ST FRACTION
  85     0038                    ACC2:   EQU     ACC1+1      ;ACCUMULATOR 2ND FRACTION
  86     0039                    ACC3:   EQU     ACC2+1      ;ACCUMULATOR 3RD FRACTION
  87     003A                    SF:     EQU     ACC3+1      ;SUBTRACTION FLAG
  88     0034                            DS      20
  89     0048                            DS      30          ;SCRATCH FOR FUNCTIONS
  90
  91                            ; CHECKS STACK THEN CALLS ROUTINE POINTED TO BY H&L
  92                            ; RECURSIVE SUBROUTINE CALL
  93     0066   CD  41  04       PUSHJ:  CALL    PCHK        ;CHECK SP
  94     0069   E9                       PCHL                ;CALL ROUTINE
  95                            ;RECURSIVE  SUBROUTINE RETURN
  96     006A   21  25  16       RETRN:  LXI     H,FRST
  97     006D   22  71  14               SHLD    PC
  98     0070   C9                       RET
  99
 100                            ; SAVE DATA ON STACK, H&L POINT TO WHATS SAVED AFTER POINT BEHIND IT
 101                            ; D,E,B,A,H,L DESTROYED
 102     0071   D1              PUSHF:  POP     D           ;SAVE RETURN ADDRESS
 103     0072   06  04                   MVI     B,WORDS     ;B IS COUNTER
 104     0074   7E              PF1:    MOV     A,M         ;GET A WORD
 105     0075   23                       INX     H
 106     0076   F5                       PUSH    PSW         ;SAVE ON STACK
 107     0077   05                       DCR     B           ;COUNT DOWN
 108     0078   C2  74  00               JNZ     PF1         ;LOOP
 109     007B   D5                       PUSH    D           ;RESTORE RETURN ADDRESS
 110     007C   C3  41  04               JMP     PCHK        ;CHECK STACK AND RETURN
```

```
STMNT    LOC   OBJECT CODE      SOURCE STATEMENT

  111
  112                          ; RESTORE DATA FROM STACK. H&L POINT TO PLACE TO PUT IT. DESTROYS A,D,E
  113    007F   D1              POPF:   POP     D       ;SAVE RETURN ADDRESS
  114    0080   01  04  00              LXI     B,WORDS ;SET UP LOOP CONTROL
  115    0083   09                      DAD     B
  116    0084   F1              POPF1:  POP     PSW     GET WORD
  117    0085   2B                      DCX     H
  118    0086   77                      MOV     M,A     ;RESTORE IT
  119    0087   0D                      DCR     C       ;COUNT DOWN
  120    0088   C2  84  00              JNZ     POPF1   ;CONTINUE
  121    008B   D5                      PUSH    D       ;RESTORE RETURN ADDRESS
  122    008C   C9                      RET
  123
  124                          ; GETS A CHARACTER FROM COMBUF ADDRESS BY AXOUT
  125                          ; CHECKS FOR TRAN. PUTS CHARACTER IN CHAR
  126                          ; NO PARAMETERS
  127                          ; UNPACK A CHARACTER. DESTROYS A,H,L,B,C,D
  128    008D   01  9E  14      GETC:   LXI     B,DMPSW ;POINT B TO DMP SWITCH
  129    0090   2A  6F  14      GETC2:  LHLD    AXOUT   ;GET BUFFER ADDRESS
  130    0093   7E                      MOV     A,M     ;GET CHAR
  131    0094   23                      INX     H       ;BUMP POINTER
  132    0095   22  6F  14              SHLD    AXOUT   ;SAVE
  133    0098   32  85  14              STA     CHAR    ;SAVE CHAR
  134    009B   FE  3F                  CPI     '?'     ;CHECK FOR ?
  135    009D   C2  B2  00              JNZ     UT2     ;NO TRACE
  136    00A0   3A  77  14              LDA     DEBGSW  ;TEST FOR TRACE ENABLED
  137    00A3   A7                      ANA     A       ;SET FLAGS
  138    00A4   C0                      RNZ             ;RET TRACE DISABLED
  139    00A5   0A                      LDAX    B       ;GET DUMP SWITCH + FLIP
  140    00A6   A7                      ANA     A       ;SET FLAG
  141    00A7   3E  00                  MVI     A,0
  142    00A9   C2  AE  00              JNZ     UT3
  143    00AC   3E  01                  MVI     A,1     ;FLIP TO 1
  144    00AE   02              UT3:    STAX    B
  145    00AF   C3  90  00              JMP     GETC2   ;GET NEXT CHARACTER INSTEAD
  146    00B2   0A              UT2:    LDAX    B       ;GET DMPSW
  147    00B3   57                      MOV     D,A     ;SAVE
  148    00B4   3A  77  14              LDA     DEBGSW  ;LOAD DEBGSW
  149    00B7   82                      ADD     D       ;AND DMPSW
  150    00B8   C0                      RNZ             ;IF BOTH ARE ZERO
  151    00B9   CD  1B  01              CALL    PRNTC   ;THEN PRINT
  152    00BC   C9                      RET
  153
  154                          ; SAVE CHAR IN BUFFER POINTED TO BY ACTIN
  155                          ; IF CHAR IS RUBOUT LAST CHAR DELETED
  156                          ; CHECKS FOR OVERFLOW
  157                          ; PACK A CHARACTER. DESTROYS H, L, A
  158    00BD   3A  85  14      PACKC:  LDA     CHAR    ;GET CHAR
  159    00C0   2A  6D  14              LHLD    AXIN    ;POINT H,L TO BUFFER
  160    00C3   FE  7F                  CPI     RUBOUT  ;IS CHAR RUBOUT?
  161    00C5   CA  D5  00              JZ      RUB1    ;JUMP IF SO
  162    00C8   77                      MOV     M,A     ;STORE CHARACTER
  163    00C9   23                      INX     H       ;BUMP POINTER
  164    00CA   22  6D  14              SHLD    AXIN    ;SAVE POINTER
  165    00CD   23                      INX     H       ;TEST GOT OVERFLOW
```

```
STMNT     LOC     OBJECT CODE       SOURCE STATEMENT

  166     00CE    3E FF             MVI     A,3770    ;BUFFER ENDS WITH 377
  167     00D0    BE                CMP     M         ;TEST
  168     00D1    CC DC 03          CZ      ERROR2    ;CALL IF ERROR (OVERFLOW)
  169     00D4    C9                RET
  170     00D5    2B          RUB1: DCX     H         ;BACKUP POINTER
  171     00D6    3E FF             MVI     A,3770    ;377 BEGIN OF BUFFER
  172     00D8    BE                CMP     M
  173     00D9    C8                RZ                ;RETURN OF BUFFER EMPTY
  174     00DA    22 6D 14          SHLD    AXIN      ;SAVE POINTER
  175     00DD    3E 5C             MVI     A,1340    ;ECHO RUBOUT
  176     00DF    CD 1B 01          CALL    PRNTC
  177     00E2    C9                RET
  178
  179                         ; ROUTINE MATCHES A CHAR WITH ONE IN A LIST AND BRANCHES
  180                         ; TO CORRESPONDING ROUTINE.  RETURNS ON A NO-MATCH.  B,C POINT
  181                         ; TO CHAR LIST.  H,L POINT TO ADDRESS LIST UPON ENTRY
  182                         ; ZERO ENDS CHAR CHECK LIST
  183                         ; SORE AND BRANCH ON AC OR CHAR, DESTROYS D
  184     00E3    A7          SORTJ:  ANA     A         ; IS AC=0?
  185     00E4    C2 EA 00          JNZ     SRT2      ;AC <>0, USE IT
  186     00E7    3A 85 14          LDA     CHAR      ;ELSE USE CHAR
  187     00EA    57          SRT2: MOV     D,A       ;SAVE CHAR
  188     00EB    0A          SRT1: LDAX    B         ;GET 1ST COMPARE CHARACTER
  189     00EC    A7                ANA     A         ;SET FLAGS
  190     00ED    C8                RZ                ;ZERO = END OF LIST, NO MATCH
  191     00EE    BA                CMP     D         ;MATCH?
  192     00EF    CA F8 00          JZ      MATCH     ;JUMP IF FOUND
  193     00F2    03                INX     B         ;CHECK NEXT CHAR
  194     00F3    23                INX     H         ;ADVANCE POINTER
  195     00F4    23                INX     H         ;TO NEXT ADDR IN TABLE
  196     00F5    C3 EB 00          JMP     SRT1      ;CONTINUE SEARCH
  197     00F8    5E          MATCH: MOV    E,M       ;MATCH FOUND
  198     00F9    23                INX     H         ;LOAD ADDRESS FROM
  199     00FA    56                MOV     D,M       ;TABLE
  200     00FB    EB                XCHG              ;AND BRANCH TO IT
  201     00FC    D1                POP     D         ;REMOVE RET ADDR FROM STACK
  202     00FD    E9                PCHL
  203
  204                         ; ROUTINE TO CHECK CHAR AGAINST A TABLE
  205                         ; B,C POINT TO TABLE
  206                         ; NUMBER STORED IN SRTCN IS DISTANCE IN TABLE
  207                         ; RETURNS TO PC+3 IF NOT FOUND
  208                         ; SORT CHAR AGAINST TABLE, DESTORYS B,C,H,L,A,D
  209     00FE    21 85 14    SORTC:  LXI     H,CHAR    ;POINT H,L TO CHAR
  210     0101    16 00             MVI     D,0       ;RESET COUNTER
  211     0103    0A          SRC2: LDAX    B         ;GET 1ST CHAR
  212     0104    A7                ANA     A         ;SET FLAGS
  213     0105    CA 16 01          JZ      SEXC      ;ZERO = END OF TABLE
  214     0108    BE                CMP     M         ;TEST
  215     0109    CA 11 01          JZ      SRC1      ;FOUND MATCH
  216     010C    14                INR     D         ;INCR COUNTER
  217     010D    03                INX     B
  218     010E    C3 03 01          JMP     SRC2      ;CHECK NEXT
  219     0111    7A          SRC1: MOV     A,D       ;GET COUNT
  220     0112    32 7C 14          STA     SRTCN     ;SAVE COUNT
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

  221    0115   C9                       RET
  222    0116   E1               SEXC:   POP     H          ;GET RET ADDRESS
  223    0117   23                       INX     H          ;CALCULATE RET ADDRESS
  224    0118   23                       INX     H          ;RETURN
  225    0119   23                       INX     H
  226    011A   E9                       PCHL               ;RETURN
  227
  228                             ; IF A=0, PRINT CHAR, ELSE PRINT A
  229                             ; PRINT AC OR CHAR, DESTROYS A
  230                                     IF      XXX
  231                             PRNTC:  ANA     A          ;SET FLAGS
  232                                     JNZ     PTC1       ;ZERO = USE CHAR
  233                                     LDA     CHAR
  234                             PTC1:   MOV     E,A        ;SAVE CHARACTER
  235                                     CPI     LF         ;DONT ECHO LINE FEED
  236                                     RZ
  237                                     CPI     CR
  238                                     JNZ     XOUTL
  239                                     IN      KYBRD
  240                                     ANI     07FH
  241                                     CPI     03H        ;C.C. BREAK
  242                                     CZ      RECOVER
  243                                     MVI     A,CR
  244                                     CALL    XOUTL
  245                                     MVI     E,LF
  246                             XOUTL:  IN      TEST       ;CRT READY?
  247                                     ANA     A
  248                                     JP      XOUTL
  249                                     MOV     A,E        ;RESTORE CHARACTER
  250                                     OUT     CRT        ;OUTPUT
  251                                     RET
  252                                     ENDF
  253                                     IF      INTEL
  254                             PRNTC:  ANA     A
  255                                     JNZ     PTC1
  256                                     LDA     CHAR
  257                             PTC1:   CPI     LF         ;DONT ECHO LINE FEED
  258                                     RZ
  259                                     CPI     CR
  260                                     JNZ     XOUTL
  261                                     CALL    3812H
  262                                     ANA     A
  263                                     JP      PTC2
  264                                     CALL    XI33
  265                                     CPI     03H        ;C.C. BREAK
  266                                     CZ      RECOVER
  267                             PTC2:   MVI     A,CR
  268                                     CALL    XOUTL
  269                                     MVI     A,RUBOUT
  270                                     CALL    XOUTL
  271                                     MVI     A,LF
  272                             XOUTL:  MOV     C,A
  273                                     CALL    3809H
  274                                     RET
  275                                     ENDF
```

```
STMNT      LOC     OBJECT CODE       SOURCE STATEMENT

  276                                          IF      PROCT
  277    011B     A7                PRNTC:    ANA     A
  278    011C     C2  22  01                   JNZ     PTC1
  279    011F     3A  85  14                   LDA     CHAR
  280    0122     FE  0A            PTC1:     CPI     LF
  281    0124     C8                          RZ
  282    0125     FE  0D                       CPI     CR
  283    0127     C2  45  01                   JNZ     XOUTL
  284    012A     CD  2E  04                   CALL    STAT
  285    012D     A7                          ANA     A
  286    012E     F2  39  01                   JP      PTC2
  287    0131     CD  36  04                   CALL    XI33
  288    0134     FE  03                       CPI     03H
  289    0136     CC  E3  03                   CZ      RECOVER
  290    0139     3E  0D            PTC2:     MVI     A,CR
  291    013B     CD  45  01                   CALL    XOUTL
  292    013E     3E  0A                       MVI     A,LF
  293    0140     CD  45  01                   CALL    XOUTL
  294    0143     3E  00                       MVI     A,0
  295                               ;   XOUTL   DRIVER   (OUTPUT)
  296                               ;
  297    0145     4F                XOUTL:    MOV     C,A       ;SAVE TEMP IN REG C
  298    0146     DB  00            TBE:      IN      0         ;CHECK TBE
  299    0148     07                          RLC               PUT IN CARRY
  300    0149     00                          NOP               ;;TO FOOL THE EXPERTS
  301    014A     D2  46  01                   JNC     TBE
  302    014D     79                          MOV     A,C
  303    014E     D3  01                       OUT     1         ;OUTPUT IT
  304    0150     C9                          RET               ;ALL DONE
  305                               ;
  306                                          ENDF
  307                                          IF      INTRP
  308                               PRNTC:     ANA     A
  309                                          JNZ     PTC1
  310                                          LDA     CHAR
  311                               PTC1:
  312                               XOUTL:     OUT     3
  313                                          RET
  314                                          ENDF
  315
  316                               ; READS A CHAR AND PRINTS ECHO IF NECESSARY
  317                               ; READ DATA INTO A CHARACTER AND PRINT IT
  318    0151     CD  36  04        READC:    CALL    XI33      ;READ A CHAR
  319    0154     32  85  14                   STA     CHAR      ;SAVE IT
  320    0157     01  3A  15                   LXI     B,ECHOLST ;POINT B TO LIST
  321    015A     CD  FE  00                   CALL    SCRTC     ;STORC
  322    015D     00                          NOP               ;RETURN + 3
  323    015E     00                          NOP
  324    015F     C9                          RET               ;YES, RETURN
  325    0160     97                          SUB     A         ;CLEAR A
  326    0161     C3  1B  01                   JMP     PRNTC     ;ECHO
  327
  328                               ; PRINTS XX.XX ACCORDING TO LINENO
  329                               ; PRINT C(LINENO)
  330    0164     3A  87  14        PRNTLN:   LDA     LINENO+1  ;GET LINENO
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

  331   0167  CD 7B 01              CALL    PRNT        ;PRINT 2 DIGITS
  332   016A  3E 2E                 MVI     A,'.'
  333   016C  CD 1B 01              CALL    PRNTC       ;PRINT "."
  334   016F  3A 86 14              LDA     LINENO
  335   0172  CD 7B 01              CALL    PRNT        ;PRINT STEP
  336   0175  3E 20                 MVI     A,SP        ;PRINT SPACE
  337   0177  CD 1B 01              CALL    PRNTC
  338   017A  C9                    RET
  339   017B  57              PRNT: MOV     C,A         ;SAVE
  340   017C  E6 F0                 ANI     0F0H        ;GET 1ST DIGIT
  341   017E  0F                    RRC
  342   017F  0F                    RRC
  343   0180  0F                    RRC
  344   0181  0F                    RRC
  345   0182  C6 30                 ADI     '0'         ;MAKE CHAR
  346   0184  FE 3A                 CPI     ':'
  347   0186  FA 8B 01              JM      $+5
  348   0189  C6 07                 ADI     'A'-'9'-1
  349   018B  CD 1B 01              CALL    PRNTC
  350   018E  7A                    MOV     A,D         ;GET 2ND
  351   018F  E6 0F                 ANI     0FH
  352   0191  C6 30                 ADI     '0'
  353   0193  FE 3A                 CPI     ':'
  354   0195  FA 9A 01              JM      $+5
  355   0198  C6 07                 ADI     'A'-'9'-1
  356   019A  CD 1B 01              CALL    PRNTC
  357   019D  C9                    RET
  358
  359                       ; RETURNS NOT FOUND
  360                       ; RERURNS+3 FOUND
  361                       ; THIS LN=FOUND LINE OR NEXT LARGER
  362                       ; LASTLN = LESSER AND/OR LAST
  363                       ; TEXTP IS SET
  364                       FINDLN:
  365   019E  21 25 16              LXI     H,OFFS
  366   01A1  22 75 14              SHLD    LASTLN      ;INIT POINTERS
  367   01A4  22 73 14        F2:   SHLD    THISLN
  368   01A7  23                    INX     H
  369   01A8  23                    INX     H           ;POINT TO # IN LINE
  370   01A9  23                    INX     H
  371   01AA  3A 87 14              LDA     LINENO+1
  372   01AD  BE                    CMP     M
  373   01AE  DA C4 01              JC      FEND3
  374   01B1  C2 CF 01              JNZ     FINDN
  375   01B4  2B              F3:   DCX     H
  376   01B5  3A 86 14              LDA     LINENO
  377   01B8  BE                    CMP     M
  378   01B9  DA C4 01              JC      FEND3
  379   01BC  C2 CF 01              JNZ     FINDN
  380   01BF  D1              F4:   POP     D
  381   01C0  13                    INX     D
  382   01C1  13                    INX     D
  383   01C2  13                    INX     D
  384   01C3  D5                    PUSH    D
  385   01C4  2A 73 14        FEND3: LHLD   THISLN
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

 396    01C7  23               INX   H
 397    01C8  23               INX   H
 398    01C9  23               INX   H
 399    01CA  23               INX   H
 400    01CB  22 6F 14         SHLD  AXOUT
 401    01CE  C9               RET
 402    01CF  2A 73 14   FINDN: LHLD  THISLN
 403    01D2  22 75 14         SHLD  LASTLN
 404    01D5  7E               MOV   A,M
 405    01D6  23               INX   H
 406    01D7  66               MOV   H,M
 407    01D8  6F               MOV   L,A
 408    01D9  B4               ORA   H
 409    01DA  CA C4 01         JZ    FEND3
 410    01DD  C3 A4 01         JMP   F2
 411
 412                           ; TERMIANTE BUFFERED LINE
 413                           ; INSERT LINE POINTERES, H=LASTLN, B=BUFR
 414    01E0  2A 82 14   ENDLN: LHLD  BUFR
 415    01E3  44               MOV   B,H
 416    01E4  4D               MOV   C,L
 417    01E5  2A 6D 14         LHLD  AXIN
 418    01E8  EB               XCHG
 419    01E9  2A 75 14         LHLD  LASTLN
 420    01EC  7E               MOV   A,M
 421    01ED  02               STAX  B
 422    01EE  23               INX   H
 423    01EF  03               INX   B
 424    01F0  7E               MOV   A,M
 425    01F1  02               STAX  B
 426    01F2  2B               DCX   H
 427    01F3  0B               DCX   B
 428    01F4  3A 82 14         LDA   BUFR
 429    01F7  77               MOV   M,A
 430    01F8  23               INX   H
 431    01F9  3A 83 14         LDA   BUFR+1
 432    01FC  77               MOV   M,A
 433    01FD  EB         E3:   XCHG
 434    01FE  22 82 14         SHLD  BUFR
 435    0201  22 7A 14         SHLD  LASTV
 436    0204  C9               RET
 437
 438                           ; NO PARAMETERS
 439                           ; DELETES LEADING SPACES FROM COMMAND
 440                           ; IGNORE SPACES AND LEADING ZEROS
 441    0205  3A 85 14   SPNOR: LDA   CHAR      ;GET CHAR
 442    0208  FE 20            CPI   ' '       ;IS IT A SPACE?
 443    020A  C0               RNZ
 444    020B  CD 8D 00         CALL  GETC      ;GET NEXT CHARACTER
 445    020E  C3 05 02         JMP   SPNOR     ;IGNORE SPACES
 446
 447                           ; GET A LINE # FROM CHARACTERS
 448                           ; IF ALL, LINENO IS
 449                           ; FORM LINENO (TWO BYTES) AS DCB CHAR LINE #
 450                           ; GROUP # IN BYTES OF LINENO, LINE # IN BYTE 2
```

```
STMNT   LOC   OBJECT CODE     SOURCE STATEMENT

  441                        ; RETURNS RESULT IN LINENO AND IN D,E
  442                        ; UNPACK AND FORM A LINE NUMBER, DESTROYS A, E
  443   0211    CD 05 02     GETLN:  CALL    SPNOR     ;IGNORE LEADING SPACES
  444   0214    3A 85 14             LDA     CHAR
  445   0217    FE 41                CPI     'A'       ;"ALL" IS SPECIAL
  446   0219    CA A5 02             JZ      ALL
  447   021C    CD B9 02             CALL    TESTN     ;TEST1
  448   021F    C3 8F 02             JMP     GZERR     ;ILLEGAL GROUP ZERO ISE
  449   0222    C3 8F 02             JMP     GZERR     ;OTHER
  450   0225    3A 85 14             LDA     CHAR
  451   0228    E6 0F                ANI     0FH       ;ISOLATE DIGIT
  452   022A    5F                   MOV     E,A       ;SAVE DIGIT
  453   022B    CD 8D 00             CALL    GETC      ;GET NEXT
  454   022E    CD B9 02             CALL    TESTN     ;TEST2
  455   0231    C3 47 02             JMP     GT1       ;PERIOD (ONE DIGIT GROUP #)
  456   0234    C3 8F 02             JMP     GZERR     ;OTHER
  457   0237    7B                   MOV     A,E       ;GET HIGH ORDER OVER DIGIT
  458   0238    07                   RLC
  459   0239    07                   RLC
  460   023A    07                   RLC
  461   023B    07                   RLC
  462   023C    5F                   MOV     E,A       ;FIX GROUP
  463   023D    3A 85 14             LDA     CHAR
  464   0240    E6 0F                ANI     0FH
  465   0242    F5                   PUSH    PSW       SAVE A
  466   0243    CD 8D 00             CALL    GETC      ;GET NEXT CHAR
  467   0246    F1                   POP     PSW
  468   0247    B3           GT1:    ORA     E
  469   0248    CA 8F 02             JZ      GZERR     ;ILLEGAL GROUP ZERO
  470   024B    32 87 14             STA     LINENO+1
  471   024E    CD B9 02             CALL    TESTN     ;TEST
  472   0251    C3 5A 02             JMP     GT2       ;PERIOD
  473   0254    C3 B6 02             JMP     GERR      ;OTHER
  474   0257    C3 8F 02             JMP     GZERR     ;TOO LARGE GROUP
  475   025A    CD 8D 00     GT2:    CALL    GETC      ;GET NEXT CHAR
  476   025D    CD B9 02             CALL    TESTN     ;TEST3
  477   0260    C3 8F 02             JMP     GZERR     ;PERIOD
  478   0263    C3 B6 02             JMP     GERR      ;OTHER
  479   0266    3A 85 14             LDA     CHAR
  480   0269    E6 0F                ANI     0FH       ;GET DIGIT
  481   026B    07                   RLC               ;MOVE OVER
  482   026C    07                   RLC
  483   026D    07                   RLC
  484   026E    07                   RLC
  485   026F    5F                   MOV     E,A       ;SAVE
  486   0270    CD 8D 00             CALL    GETC      ;READ LAST CHAR
  487   0273    CD B9 02             CALL    TESTN     ;TEST4
  488   0276    C3 B6 02             JMP     GERR      ;PERIOD
  489   0279    C3 92 02             JMP     GT4       ;OTHER
  490   027C    3A 85 14             LDA     CHAR      ;DIGIT
  491   027F    E6 0F                ANI     0FH
  492   0281    B3           GT3:    ORA     E
  493   0282    5F                   MOV     E,A
  494   0283    CD 8D 00             CALL    GETC      ;CHECK SIZE
  495   0286    CD B9 02             CALL    TESTN     ;TEST
```

```
STMNT    LOC     OBJECT CODE      SOURCE S

 496     0289    C3  B6  02               JMP     GERR        ;PERIOD
 497     028C    C3  92  02               JMP     GT4         ;OK
 498     028F    CD  DC  03       GZERR:  CALL    ERROR2      ;OTHER :ILLEGAL GROUP ZERO
 499     0292    7B               GT4:    MOV     A,E
 500     0293    32  86  14               STA     LINENO      ;SAVE
 501     0296    A7                       ANA     A
 502     0297    CA  A0  02               JZ      GROUP       ;STEP IS 0
 503     029A    3E  80                   MVI     A,2000      ;SETP
 504     029C    32  84  14               STA     MAGSW
 505     029F    C9                       RET
 506     02A0    97               GROUP:  SUB     A           ;GROUP
 507     02A1    32  84  14               STA     MAGSW
 508     02A4    C9                       RET
 509     02A5    3E  01           ALL:    MVI     A,1
 510     02A7    32  84  14               STA     MAGSW       ;ALL
 511     02AA    3E  7F                   MVI     A,RUBOUT
 512     02AC    32  85  14               STA     CHAR
 513     02AF    21  01  00               LXI     H,1
 514     02B2    22  86  14               SHLD    LINENO
 515     02B5    C9                       RET
 516     02B6    CD  DC  03       GERR:   CALL    ERROR4      ;BAD LINE #
 517
 518                              ; RETURNS IF PERIOD
 519                              ; RETURNS+3 IF NON-DIGIT
 520                              ; RETURNS+6 IF DIGIT
 521                              ; NO PARAMETERS
 522                              ; TEST FOR PERIOD, OTHER, NUMBER DESTROYS H,L,B,C,A
 523     02B9    01  03  00       TESTN:  LXI     B,3         ;PC INCREMENT
 524     02BC    E1                       POP     H           ;GET RETURN ADDRESS
 525     02BD    3A  85  14               LDA     CHAR        ;GET CHARACTER TO TEST
 526     02C0    FE  2E                   CPI     '.'         ;TEST FOR PERIOD
 527     02C2    CA  D1  02               JZ      B1
 528     02C5    FE  3A                   CPI     3AH
 529     02C7    F2  D0  02               JP      B3
 530     02CA    FE  30                   CPI     30H
 531     02CC    FA  D0  02               JM      B3
 532     02CF    09                       DAD     B           ;YES
 533     02D0    09               B3:     DAD     B
 534     02D1    97               B1:     SUB     A           ;CLEAR AC
 535     02D2    E9                       PCHL
 536
 537                              ; TEST SORTCN FOR LEFT PAREN
 538                              ; RETURN+3 IF SO
 539                              ; SKIP IF S<SORTCN<11 (I.E. ANL.PAR)
 540     02D3    3A  7C  14       TSTLPR: LDA     SRTCN
 541     02D6    47                       MOV     B,A
 542     02D7    D6  09                   SUI     9           ;<12?
 543     02D9    F0                       RP
 544     02DA    E1                       POP     H           ;GET RETURN ADDRESS
 545     02DB    78                       MOV     A,B
 546     02DC    D6  06                   SUI     6           ;>5?
 547     02DE    FA  E5  02               JM      T2          ;NO
 548     02E1    23                       INX     H
 549     02E2    23                       INX     H
 550     02E3    23                       INX     H
```

```
STMNT   LOC   OBJECT CODE     SOURCE STATEMENT

  551   02E4  E9              PCHL
  552   02E5  97          T2: SUB   A
  553   02E6  E9              PCHL
  554
  555                   ; SKIP IF G(AC)=G(LINENO)
  556   02E7  47        TSTGRP: MOV   B,A
  557   02E8  3A 87 14        LDA   LINENO+1
  558   02EB  B8              CMP   B
  559   02EC  78              MOV   A,B
  560   02ED  C0              RNZ
  561   02EE  E3              XTHL
  562   02EF  23              INX   H
  563   02F0  23              INX   H
  564   02F1  23              INX   H
  565   02F2  E3              XTHL
  566   02F3  C9              RET
  567
  568                   ; TEST THE NATURE OF AN ALPHABETIC
  569                   ; RETURNS TERM
  570                   ; RETURN+3 NUMBER
  571                   ; RETURN+6 F
  572                   ; RETURN+9 ALPHANUMERIC
  573                   ; TERM: NUMBER: FUNCTION: LETTER AND IGNORE SPACES
  574   02F4  C5        TESTC:  PUSH  B
  575   02F5  D5              PUSH  D
  576   02F6  CD 05 02        CALL  SPNOR     ;IGNORE SPACES
  577   02F9  01 3D 15        LXI   B,TERMS   ;TEST TERMINATORS
  578   02FC  CD FE 00        CALL  SORTC
  579   02FF  D1              POP   D
  580   0300  C1              POP   B
  581   0301  C9              RET             ;SORTCN IS SET
  582   0302  D1              POP   D
  583   0303  C1              POP   B
  584   0304  E1              POP   H         ;NOT TERM
  585   0305  23              INX   H
  586   0306  23              INX   H
  587   0307  23              INX   H
  588   0308  3A 85 14        LDA   CHAR
  589   030B  FE 46           CPI   'F'       ;TEST FOR 'F'
  590   030D  CA 25 03        JZ    XT3       ;YES
  591   0310  E5              PUSH  H
  592   0311  C5              PUSH  B
  593   0312  D5              PUSH  D
  594   0313  CD B9 02        CALL  TESTN
  595   0316  D1              POP   D
  596   0317  C1              POP   B
  597   0318  C9              RET             ;PERIOD
  598   0319  C3 1F 03        JMP   XT2       ;OTHER
  599   031C  D1              POP   D
  600   031D  C1              POP   B
  601   031E  C9              RET             ;NUMBER
  602   031F  D1        XT2:  POP   D
  603   0320  C1              POP   B
  604   0321  E1              POP   H
  605   0322  23              INX   H
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

 606    0323    23               INX   H
 607    0324    23               INX   H
 608    0325    23               INX   H
 609    0326    23               INX   H
 610    0327    23               INX   H
 611    0328    E9               PCHL                ;RET: T:N:F:A
 612
 613                             ;UNCHAIN A LINE, RECOVER SPACE
 614                             ;REMOVE OLD LINE OF TEXT
 615    0329    CD 9E 01         DELETE: CALL  FINDLN  ;FIND THE LINE
 616    032C    C9               RET                 ;ALREADY GONE
 617    032D    00               NOP
 618    032E    00               NOP
 619    032F    97               SUB   A
 620    0330    32 7B 15         STA   TEMP+1
 621    0333    1E 04            MVI   E,4           ;COUNT POINTER AND LINE NUMBER
 622    0335    3E 01            MVI   A,1           ;DISABLE TRACE
 623    0337    32 77 14         STA   DEBGSW
 624    033A    CD 8D 00         D1:   CALL  GETC    ;MEASURE LENGTH OF LINE TO DELETE
 625    033D    1C               INR   E
 626    033E    3A 85 14         LDA   CHAR
 627    0341    FE 0D            CPI   CR
 628    0343    C2 3A 03         JNZ   D1
 629    0346    7B               MOV   A,E
 630    0347    32 7A 15         STA   TEMP          ;SAVE COUNT
 631    034A    2F               CMA                 ;AND NEGATIVE COUNT
 632    034B    3C               INR   A
 633    034C    32 7F 14         STA   CNTR
 634    034F    2A 73 14         LHLD  THISLN        ;CHECK FOR FINISHED
 635    0352    23               INX   H
 636    0353    23               INX   H             ;POINT TO LINE #
 637    0354    7E               MOV   A,M
 638    0355    23               INX   H
 639    0356    B6               ORA   M             ;CHECK FOR LINE ZERO
 640    0357    CA 64 04         JZ    START         ;IGNORE LINE ZERO DELETE
 641    035A    2A 73 14         LHLD  THISLN
 642    035D    44               MOV   B,H           ;SET UP POINTER
 643    035E    4D               MOV   C,L
 644    035F    2A 75 14         LHLD  LASTLN        ;DISCONNECT
 645    0362    0A               LDAX  B
 646    0363    77               MOV   M,A
 647    0364    23               INX   H
 648    0365    03               INX   B
 649    0366    0A               LDAX  B
 650    0367    0B               DCX   B
 651    0368    77               MOV   M,A
 652    0369    21 25 16         LXI   H,CFRS        ;START AT LINE 0
 653    036C    5E               DOK:  MOV   E,M     ;GET NEXT LINE
 654    036D    7B               MOV   A,E
 655    036E    23               INX   H
 656    036F    56               MOV   D,M           ;D,E POINT TO NEXT LINE
 657    0370    B2               ORA   D
 658    0371    CA 9F 03         JZ    DONE          ;CHECK FOR FINISHED
 659    0374    2B               DCX   H
 660    0375    7A               MOV   A,D
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

 661    0376    B8               CMP     B
 662    0377    DA  98  03       JC      D2          ;THISLN >?
 663    037A    C2  82  03       JNZ     D3          ;NOT EQUAL
 664    037D    7B               MOV     A,E
 665    037E    B9               CMP     C           ;TEST LOW ORDER
 666    037F    DA  98  03       JC      D2          ;GREATER
 667    0382    E5           D3: PUSH    H           ;SAVE POINTER
 668    0383    2A  7F  14       LHLD    CNTR        ;GET DISPLACEMENT
 669    0386    19           D4: DAD     D           ;ADD DISP<0
 670    0387    22  7C  15       SHLD    TEMP2       ;SAVE
 671    038A    E1               POP     H           ;RESTORE POINTER
 672    038B    3A  7C  15       LDA     TEMP2       ;GET NEW POINTER
 673    038E    77               MOV     M,A         ;REPLACE OLD
 674    038F    23               INX     H
 675    0390    3A  7D  15       LDA     TEMP2+1
 676    0393    77               MOV     M,A
 677    0394    EB               XCHG                ;GET NEXT
 678    0395    C3  6C  03       JMP     DOK         ;CONTINUE
 679    0398    E5           D2: PUSH    H           ;THISLN>  ;SAVE POINTER
 680    0399    21  00  00       LXI     H,0         ;DISP IS 0
 681    039C    C3  86  03       JMP     D4
 682
 683                             ; GARBAGE COLLECTION (LINE DELETED)
 684    039F    2A  73  14   DONE: LHLD    THISLN      ;GARBAGE COLLECTION
 685    03A2    E5               PUSH    H           ;XRT
 686    03A3    EB               XCHG
 687    03A4    2A  7A  15       LHLD    TEMP        ;GET DISP
 688    03A7    19               DAD     D           ;ADD
 689    03A8    E5               PUSH    H           ;XRT2
 690    03A9    2A  82  14       LHLD    BUFR
 691    03AC    EB               XCHG
 692    03AD    2A  7F  14       LHLD    CNTR
 693    03B0    19               DAD     D           ;CALCULATE NEW END OF BUFR
 694    03B1    22  82  14       SHLD    BUFR
 695    03B4    2A  6D  14       LHLD    AXIN
 696    03B7    7C               MOV     A,H
 697    03B8    2F               CMA                 ;CALCULATE COUNT FOR LINE MOVEMENT
 698    03B9    67               MOV     H,A
 699    03BA    7D               MOV     A,L
 700    03BB    2F               CMA
 701    03BC    6F               MOV     L,A
 702    03BD    D1               POP     D           ;XRT2
 703    03BE    D5               PUSH    D           ;SAVE
 704    03BF    19               DAD     D
 705    03C0    44               MOV     B,H
 706    03C1    4D               MOV     C,L         ;B,C HOLD COUNT
 707    03C2    2A  6D  14       LHLD    AXIN
 708    03C5    EB               XCHG
 709    03C6    2A  7F  14       LHLD    CNTR
 710    03C9    19               DAD     D           ;CALCULATE NEW AXIN POINTER
 711    03CA    22  6D  14       SHLD    AXIN
 712    03CD    D1               POP     D           ;XRT2
 713    03CE    E1               POP     H           ;XRT
 714    03CF    1A           DN1: LDAX    D           ;MEMORY SHIFT LOOP
 715    03D0    77               MOV     M,A
```

```
STMNT    LOC     OBJECT CODE      SOURCE STATEMENT

  716    03D1    03               INX     B
  717    03D2    23               INX     H
  718    03D3    13               INX     D
  719    03D4    78               MOV     A,B         ;FINISHED?
  720    03D5    B1               ORA     C
  721    03D6    C2  CF  03       JNZ     DN1         ;NO
  722    03D9    C3  29  03       JMP     DELETE
  723
  724                           ; ERRORS
  725    03DC                    ERROR5: EQU     $
  726    03DC    E1              ERR2:   POP     H           ;GET ERROR NUMBER
  727    03DD    22  86  14              SHLD    LINENO
  728    03E0    C3  E9  03              JMP     R3
  729    03E3    21  00  00      RECOVER: LXI    H,0         ;SET UP INIT COUNTER
  730    03E6    22  86  14              SHLD    LINENO
  731    03E9    06  20          R3:     MVI     B,400
  732    03EB    21  AB  15              LXI     H,OPTRO     ;FIX I/O BUFFERS
  733    03EE    22  6D  14              SHLD    AXIN
  734    03F1    36  00          R2:     MVI     M,0         ;CLEAR BUFFER
  735    03F3    23                      INX     H
  736    03F4    05                      DCR     B
  737    03F5    C2  F1  03              JNZ     R2
  738    03F8    2A  03  00              LHLD    SPAD
  739    03FB    F9                      SPHL
  740    03FC    3E  0D                  MVI     A,CR
  741    03FE    CD  1B  01              CALL    PRNTC
  742    0401    3E  3F                  MVI     A,'?'
  743    0403    CD  1B  01              CALL    PRNTC       ;PRINT
  744    0406    CD  64  01              CALL    PRNTLN      ;PRINT ERROR
  745    0409    2A  71  14              LHLD    PC
  746    040C    23                      INX     H
  747    040D    23                      INX     H
  748    040E    7E                      MOV     A,M
  749    040F    23                      INX     H
  750    0410    66                      MOV     H,M
  751    0411    6F                      MOV     L,A
  752    0412    B4                      ORA     H           ;AND LINE # UNLESS ZERO
  753    0413    CA  26  04              JZ      R4
  754    0416    22  86  14              SHLD    LINENO      ;GET LOC. OF ERROR
  755    0419    3E  40                  MVI     A,'@'       ;PRINT @
  756    041B    CD  1B  01              CALL    PRNTC
  757    041E    3E  20                  MVI     A,SP
  758    0420    CD  1B  01              CALL    PRNTC       ;PRINT SPACE
  759    0423    CD  64  01              CALL    PRNTLN      ;PRINT LINE @
  760    0426    3E  0D          R4:     MVI     A,CR
  761    0428    CD  1B  01              CALL    PRNTC
  762    042B    C3  64  04              JMP     START
  763
  764                           ; READS KEYBOARD, RETURNS CHARACTER IN A
  765                           ; INPUT WAIT
  766                                   IF      XXX
  767                           XI33:   IN      KYBRD       ;READ KEYBOARD
  768                                   ANA     A           ;SET FLAGS
  769                                   JP      XI33        ;NOTHING READ
  770                                   ANI     7FH         ;STRIP PARITY
```

```
STMNT     LOC     OBJECT CODE       SOURCE STATEMENT

  771                                         RET
  772                                         ENDF
  773                                         IF        INTEL
  774                               XI33:     CALL      3803H
  775                                         ANI       7FH
  776                                         RET
  777                                         ENDF
  778                                         IF        INTRP
  779                               XI33:     IN        0
  780                                         ANI       7FH
  781                                         RET
  782                                         ENDF
  783                               ;
  784                                         IF        PROCT
  785     042E      DB  00          STAT:     IN        0           ;GET DEV STATUS
  786     0430      E6  40                    ANI       40H
  787     0432      C8                        RZ                    ;RETURN IF ZERO (NO CHAR)
  788     0433      97                        SUB       A           ;MAKE IT FF
  789     0434      3D                        DCR       A
  790     0435      C9                        RET
  791                               XI33:
  792     0436      CD  2E  04      IN8:      CALL      STAT
  793     0439      CA  36  04                JZ        IN8
  794     043C      DB  01                    IN        1
  795     043E      E6  7F                    ANI       127
  796     0440      C9                        RET
  797                               ;
  798                               ;
  799                                         ENDF
  800
  801                               ; CHECK STACK FOR OVERFLOW, H,L,D,E,A DESTROYED
  802     0441      E5              PCHK:     PUSH      H           ;SAVE H&L
  803     0442      D5                        PUSH      D           ;SAVE D&E
  804     0443      21  00  00                LXI       H,0         ;CLEAR H,I
  805     0446      39                        DAD       SP          ;GET SP
  806     0447      EB                        XCHG
  807     0448      2A  7A  14                LHLD      LASTV       ¬SUBSTRACT
  808     044B      7C                        MOV       A,H
  809     044C      2F                        CMA
  810     044D      67                        MOV       H,A
  811     044E      7D                        MOV       A,L
  812     044F      2F                        CMA
  813     0450      6F                        MOV       L,A
  814     0451      23                        INX       H
  815     0452      19                        DAD       D           ;LOWEST LOCATION
  816     0453      7C                        MOV       A,H
  817     0454      A7                        ANA       A           ;TEST
  818     0455      FC  DC  03                CM        ERROR5
  819     0458      C2  61  04                JNZ       PCHK2
  820     045B      B5                        ORA       L
  821     045C      D6  0A                    SUI       120         ;WITHIN TEN STACK LOCATIONS?
  822     045E      FC  DC  03                CC        ERROR3      ;IF < 0 THE STACK OVERFLOW
  823     0461      D1              PCHK2:    POP       D
  824     0462      E1                        POP       H
  825     0463      C9                        RET       ;STACK OK
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

  826
  827                          ; COMMAND INPUT DRIVER
  828    0464    21  64  15     START:  LXI   H,CFRSX   ;GET PC =>0
  829    0467    22  71  14             SHLD  PC
  830    046A    3E  C3                 MVI   A,0C3H    ;SET UP RESTART AT 0
  831    046C    32  00  00             STA   0
  832    046F    21  F3  03             LXI   H,RECOVER
  833    0472    22  01  00             SHLD  1
  834    0475    97                     SUB   A
  835    0476    32  33  00             STA   OVER+SCRB ;CLEAR OVERFLOW
  836    0479    32  8E  14             STA   LIST3+1   ;RESET MODIFY
  837    047C    32  77  14             STA   DEBGSW
  838    047F    3E  01                 MVI   A,1
  839    0481    32  9E  14             STA   DMPSW     ;INIT UNPACK AND TRACE SWITCH
  840    0484    2A  03  00     IBAR:   LHLD  SPAD      ;GET STACK START LOC
  841    0487    F9                     SPHL  ;SET STACK POINTER
  842    0488    21  D4  15             LXI   H,COMBUF
  843    048B    22  6D  14             SHLD  AXIN
  844    048E    3E  2A                 MVI   A,'*'
  845    0490    CD  1B  01             CALL  PRNTC
  846    0493    CD  51  01     IGNOR:  CALL  READC
  847    0496    97                     SUB   A
  848    0497    01  8A  14             LXI   B,LIST7
  849    049A    21  C4  14             LXI   H,INLIST
  850    049D    CD  E3  00             CALL  SORTJ
  851    04A0    CD  BD  00             CALL  PACKC
  852    04A3    C3  93  04             JMP   IGNOR
  853
  854                          ; COMMAND INPUT PROCESSOR
  855    04A6    CD  BD  00     IRETN:  CALL  PACKC     ;PUT CR IN BUFFER
  856    04A9    97                     SUB   A
  857    04AA    CD  1B  01             CALL  PRNTC
  858    04AD    21  D4  15             LXI   H,COMBUF  ;INITIALIZE TEXTP
  859    04B0    22  6F  14     GONE:   SHLD  AXOUT     ;SAVE IN POINTER
  860    04B3    CD  8D  00             CALL  GETC      ;READ 1ST CHARACTER
  861    04B6    2A  03  00             LHLD  SPAD      ;GET STACK START LOC
  862    04B9    F9                     SPHL  ;SET STACK POINTER
  863    04BA    CD  05  02             CALL  SPNOR     ;IGNORE LEADING BLANKS
  864    04BD    CD  B9  02             CALL  TESTN     ;IS THERE A LINE #
  865    04C0    C3  8F  02             JMP   GZERR     ;PERIOD, ILLEGAL 1ST CHARACTER
  866    04C3    C3  04  05             JMP   INPTX     ;IMMEDAITE COMMAND
  867    04C6    3E  01                 MVI   A,1       ;LIEN #
  868    04C8    32  77  14             STA   DEBGSW    ;DISABLE TRACE
  869    04CB    CD  11  02             CALL  GETLN     ;READ LINE NUMBER
  870    04CE    3A  84  14             LDA   NAGSW     ;TEST FOR SINGLE LINE
  871    04D1    FE  80                 CPI   2000
  872    04D3    C4  DC  03             CNZ   ERROR3    ;BAD LINE
  873    04D6    2A  82  14             LHLD  BUFR      ;SET POINTERS
  874    04D9    23                     INX   H
  875    04DA    23                     INX   H
  876    04DB    EB                     XCHG
  877    04DC    2A  86  14             LHLD  LINENO    ;SAVE LINE #
  878    04DF    EB                     XCHG
  879    04E0    73                     MOV   M,E
  880    04E1    23                     INX   H
```

```
STMNT    LOC     OBJECT CODE      SOURCE STATEMENT

  881    04E2    72                       MOV     M,D
  882    04E3    23                       INX     H
  883    04E4    22  6D  14                SHLD    AXIN
  884    04E7    CD  05  02                CALL    SPNOR       ;IGNORE SPACES AFTER LINE #
  885    04EA    C3  F0  04                JMP     SRETN
  886    04ED    CD  8D  00                CALL    GETC        ;READ 1ST CHAR AFTER LINE #
  887    04F0    CD  BD  00       SRETN:   CALL    PACKC       ;SAVE TEXT
  888    04F3    3A  85  14                LDA     CHAR
  889    04F6    FE  0D                    CPI     CR          ;TEST END OF LINE
  890    04F8    C2  ED  04                JNZ     SRETN-3     ;NOT END
  891    04FB    CD  29  03                CALL    DELETE      ;REMOVE OL LINE IF ANY
  892    04FE    CD  E0  01                CALL    ENDLN       ;INSERT NEW LINE
  893    0501    C3  64  04                JMP     START       ;REINIT POINTERS
  894    0504    21  B7  05       INPTX:   LXI     H,PROC      ;CALL PROC
  895    0507    CD  66  00                CALL    PUSHJ
  896    050A    2A  71  14                LHLD    PC          ;CHECK NEXT LINE
  897    050D    7E                        MOV     A,M         ;END OF PROGRAM?
  898    050E    5F                        MOV     E,A
  899    050F    23                        INX     H
  900    0510    56                        MOV     D,M
  901    0511    B6                        ORA     M
  902    0512    EB                        XCHG
  903    0513    CA  64  04                JZ      START       ;YES
  904    0516    22  71  14                SHLD    PC
  905    0519    23                        INX     H
  906    051A    23                        INX     H
  907    051B    23                        INX     H
  908    051C    23                        INX     H
  909    051D    C3  B0  04                JMP     GONE        ;PROCESS NEXT COMMAND
  910
  911                              ; RECURSIZE OPERATE, EXECUTE OR CALL
  912    0520    CD  11  02       DO:      CALL    GETLN       ;EXECUTE ONE LINE, GROUP OR ALL
  913    0523    2A  71  14                LHLD    PC
  914    0526    E5                        PUSH    H           ;SAVE PC
  915    0527    2A  6F  14                LHLD    TEXTP
  916    052A    E5                        PUSH    H
  917    052B    21  84  14       DGRP:    LXI     H,NAGSW     ;SAVE NAGSW, CHAR, LINENO
  918    052E    CD  71  00                CALL    PUSHF
  919    0531    3A  84  14                LDA     NAGSW       ;CHECK SWITCH
  920    0534    A7                        ANA     A
  921    0535    FA  88  05                JM      DOONE       ;ONE LINE
  922    0538    CD  9E  01                CALL    FINDLN      ;INIT FOR GROUP, SET THISLN
  923    053B    00                        NOP
  924    053C    00                        NOP
  925    053D    00                        NOP
  926    053E    2A  73  14                LHLD    THISLN
  927    0541    22  71  14                SHLD    PC
  928    0544    23                        INX     H
  929    0545    23                        INX     H
  930    0546    23                        INX     H
  931    0547    7E                        MOV     A,M         ;SET GROUP NO
  932    0548    2B                        DCX     H
  933    0549    CD  E7  02                CALL    TSTGRP      ;CHECK VALIDITY
  934    054C    CD  DC  03                CALL    ERROR2      ;NO SUCH GROUP
  935    054F    21  B4  05       DGRP1:   LXI     H,PROCESS   ;PROCESS COMMAND
```

```
STMNT     LOC     OBJECT CODE       SOURCE STATEMENT

 936      0552    CD 66 00                  CALL    PUSHJ
 937      0555    21 84 14                  LXI     H,NAGSW
 938      0558    CD 7F 00                  CALL    POPF        ;RESTORF
 939      055B    2A 71 14                  LHLD    PC          ;CHECK EOT
 940      055E    7E                        MOV     A,M
 941      055F    23                        INX     H
 942      0560    B6                        ORA     M
 943      0561    CA 9A 05                  JZ      DCONT       ;DONE
 944      0564    2B                        DCX     H
 945      0565    5E                        MOV     E,M
 946      0566    23                        INX     H
 947      0567    56                        MOV     D,M
 948      0568    EB                        XCHG
 949      0569    23                        INX     H
 950      056A    23                        INX     H           ;POINT TO LINENO
 951      056B    3A 84 14                  LDA     NAGSW       ;CHECK FOR GROUP
 952      056E    A7                        ANA     A
 953      056F    CA 75 05                  JZ      DR3
 954      0572    F2 7E 05                  JP      DR2         ;DO ALL
 955      0575    23              DR3:      INX     H
 956      0576    7E                        MOV     A,M
 957      0577    2B                        DCX     H
 958      0578    CD E7 02                  CALL    TSTGRP
 959      057B    C3 9A 05                  JMP     DCONT       ;NOT IN GROUP
 960      057E    5E              DR2:      MOV     E,M         ;GET NEXT LINENO
 961      057F    23                        INX     H
 962      0580    56                        MOV     D,M
 963      0581    EB                        XCHG
 964      0582    22 86 14                  SHLD    LINENO
 965      0585    C3 2B 05                  JMP     DGRP        ;CONTINUE SUBROUTINE
 966      0588    CD 9E 01        DOONE:    CALL    FINDLN      ;FIND THE LINE
 967      058B    CD DC 03                  CALL    ERROR2      ;NO SUCH LINE
 968      058E    21 B4 05                  LXI     H,PROCESS   ;EXECUTE IT
 969      0591    CD 66 00                  CALL    PUSHJ
 970      0594    21 84 14                  LXI     H,NAGSW     ;RESTORE CHAR
 971      0597    CD 7F 00                  CALL    POPF
 972      059A    E1              DCONT:    POP     H           ;RESTORE TEXT POINTERS
 973      059B    22 6F 14                  SHLD    TEXTP
 974      059E    E1                        POP     H           ;RESTORE PC
 975      059F    22 71 14                  SHLD    PC
 976      05A2    C3 B7 05                  JMP     PROC        ;CONTINUE PROCESSING
 977
 978                             ; PRIMARY CONTROL AND TRANSFER
 979      05A5    CD 11 02        GOTO:     CALL    GETLN       ;READ LINE @
 980      05A8    CD 9E 01                  CALL    FINDLN      ;LOCATE IT
 981      05AB    CD DC 03                  CALL    ERROR2      ;NOT THERE
 982      05AE    2A 73 14                  LHLD    THISLN      ;SET PC
 983      05B1    22 71 14                  SHLD    PC
 984      05B4    CD 8D 00        PROCESS:  CALL    GETC        ;TEST END OF LINE
 985      05B7    3A 85 14        PROC:     LDA     CHAR        ;FIRST CHAR READY = USE PROC
 986      05BA    FE 0D                     CPI     CR
 987      05BC    C2 C0 05                  JNZ     PC2
 988      05BF    C9              PC1:      RET                 ;EXIT PROCESS
 989      05C0    01 1F 15        PC2:      LXI     B,GLIST     ;IGNORE SPACE AND ;
 990      05C3    CD FE 00                  CALL    SORTC
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

  991    05C6   C3 B4 05                JMP     PROCESS
  992    05C9   3A 85 14                LDA     CHAR        ;SAVE COMMAND CHARACTER
  993    05CC   F5                      PUSH    PSW
  994    05CD   CD 8D 00         P4:    CALL    GETC        ;GO TO PERMINATOR
  995    05D0   01 1F 15                LXI     B,GLIST
  996    05D3   CD FE 00                CALL    SORTC
  997    05D6   C3 DC 05                JMP     PC3
  998    05D9   C3 CD 05                JMP     P4
  999                           PC3:
 1000                                    IF      INTEL
 1001                                    CALL    3812H       ;CHECK FOR C.C. BREAK
 1002                                    ANA     A
 1003                                    JP      PC3A        ;CNO CHAR AVAIL
 1004                                    CALL    XI33        ;GET CHAR
 1005                                    CPI     03H         ;C.C.
 1006                                    CZ      RECOVER     ;BREAK
 1007                                    ENDF
 1008                                    IF      XXX
 1009                                    IN      KYBRD       ;C.C. BREAK CHECK
 1010                                    ANI     07FH
 1011                                    CPI     03H         ;C.C.
 1012                                    CZ      RECOVER     ;BREAK
 1013                                    ENDF
 1014                                    IF      PROCT
 1015   05DC   CD 2E 04                  CALL    STAT
 1016   05DF   A7                        ANA     A
 1017   05E0   F2 EB 05                  JP      PC3A
 1018   05E3   CD 36 04                  CALL    XI33
 1019   05E6   FE 03                     CPI     03H
 1020   05E8   CC E3 03                  CZ      RECOVER
 1021                                    ENDF
 1022   05EB   F1              PC3A:    POP     PSW
 1023   05EC   01 DE 14                  LXI     B,COMLST    ;DO COMMAND
 1024   05EF   21 EE 14                  LXI     H,COMGO
 1025   05F2   CD E3 00                  CALL    SORTJ
 1026   05F5   CD DC 03                  CALL    ERROR2      ;ILLEGAL COMMAND
 1027
 1028                           ; OUTPUT COMMAND TEXT
 1029   05F8   3E 01            WRITE:   MVI     A,1
 1030   05FA   32 77 14                  STA     DEBGSW      ;DISABLE TRACE
 1031   05FD   CD 11 02                  CALL    GETLN       ;GET LINENO
 1032   0600   CD 9E 01         W5:      CALL    FINDLN      ;SEARCH FOR LINE
 1033   0603   C3 4A 06                  JMP     WTESTG      ;NOT THERE OR GROUP
 1034   0606   2A 86 14                  LHLD    LINENO
 1035   0609   7C                        MOV     A,H
 1036   060A   B5                        ORA     L
 1037   060B   CA 11 06                  JZ      W3
 1038   060E   CD 64 01                  CALL    PRNTLN      ;PRINT LINE # AND Q SP
 1039   0611   CD 8D 00         W3:      CALL    GETC
 1040   0614   97                        SUB     A
 1041   0615   CD 1B 01                  CALL    PRNTC       ;PRINT LINE TEXT
 1042   0618   3A 85 14                  LDA     CHAR
 1043   061B   FE 0D                     CPI     CR          ;TEST END OF LINE
 1044   061D   C2 11 06                  JNZ     W3
 1045   0620   2A 73 14                  LHLD    THISLN
```

```
STMNT     LOC     OBJECT CODE       SOURCE STATEMENT

1046      0623    7E                          MOV     A,M
1047      0624    5F                          MOV     E,A
1048      0625    23                          INX     H
1049      0626    56                          MOV     D,M
1050      0627    B6                          ORA     M           ;TEST END OF TEXT
1051      0628    EB                          XCHG
1052      0629    CA  52  06        WTST2:    JZ      W6          ;EXIT, DO NEXT ????? LINE
1053      062C    23                          INX     H
1054      062D    23                          INX     H           ;POINT TO LINENO OF NEXT
1055      062E    3A  84  14                  LDA     NAGSW
1056      0631    A7                          ANA     A
1057      0632    FA  38  06                  JM      W4
1058      0635    23                          INX     H
1059      0636    7E                          MOV     A,M
1060      0637    2B                          DCX     H
1061      0638    CD  E7  02        W4:       CALL    TSTGRP      ;TRY NEXT LINENO FOR GROUP
1062      063B    C3  56  06                  JMP     WX
1063      063E    EB                WALL:     XCHG                ;SET LINENO
1064      063F    1A                          LDAX    D
1065      0640    6F                          MOV     L,A
1066      0641    13                          INX     D
1067      0642    1A                          LDAX    D
1068      0643    67                          MOV     H,A
1069      0644    22  86  14                  SHLD    LINENO
1070      0647    C3  00  06                  JMP     W5
1071      064A    2A  73  14        WTESTG:   LHLD    THISLN      ;INIT GROUP PRINTOUT
1072      064D    7C                          MOV     A,H
1073      064E    B5                          ORA     L
1074      064F    C3  29  06                  JMP     WTST2
1075      0652    32  77  14        W6:       STA     DEBGSW
1076      0655    C9                          RET
1077      0656    3A  84  14        WX:       LDA     NAGSW
1078      0659    A7                          ANA     A
1079      065A    FA  52  06                  JM      W6
1080      065D    CA  52  06                  JZ      W6
1081      0660    97                          SUB     A
1082      0661    CD  1B  01                  CALL    PRNTC       ;PRINT CR
1083      0664    C3  3E  06                  JMP     WALL
1084
1085                              ; COMPUTED TRANSFER
1086      0667    CD  05  02        JUMP:     CALL    SPNOR       ;IGNORE VBLANKS
1087      066A    CD  B2  09                  CALL    EVAL        ;EVALUATE INSIDES
1088      066D    1E  20                      MVI     E,32
1089      066F    CD  8E  12                  CALL    FIX         ;TAKE INTEGER
1090      0672    7A                          MOV     A,D
1091      0673    C3  93  06                  JMP     I3          ;USE IF TO BARANCH
1092
1093                              ; CONDITIONAL TRANSFER PROCESS
1094      0676    CD  F4  02        IF:       CALL    TESTC       ;IGNORE SPACES AND TEST
1095      0679    CD  B2  09                  CALL    EVAL        ;T
1096      067C    00                          NOP                 ;N= DUMP THE (EFOPE
1097      067D    1E  00                      MVI     E,0
1098      067F    01  00  00                  LXI     B,0         ;F--SKIP
1099      0682    CD  D1  0F                  CALL    TST
1100      0685    3E  00                      MVI     A,0
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

1101    0687    CA  92  06              JZ      IZ
1102    068A    F2  91  06              JP      IP
1103    068D    2A  F6  14      IF3:    LHLD    COMGO+8 ;TRANSFER
1104    0690    E9                      PCHL
1105    0691    3C              IP:     INR     A
1106    0692    3C              IZ:     INR     A
1107    0693    32  7A  15      I3:     STA     TEMP
1108    0696    97                      SUB     A
1109    0697    01  20  15              LXI     B,TLIST
1110    069A    21  D8  14              LXI     H,ILIST
1111    069D    CD  E3  00              CALL    SORTJ   ;SEARCH FOR ,;CR
1112    06A0    CD  8D  00              CALL    GETC
1113    06A3    C3  96  06              JMP     I3+3
1114    06A6    CD  8D  00      IF1:    CALL    GETC
1115    06A9    3A  7A  15              LDA     TEMP
1116    06AC    3D                      DCR     A
1117    06AD    C2  93  06              JNZ     I3
1118    06B0    C3  8D  06              JMP     IF3
1119
1120                            ; LOOP CONTROL AND ASSIGNMENT
1121                            ; FOR AND SET
1122                            ; SET IS FOR WITH NO LOOP CONTROL
1123                            SET:
1124    06B3    21  B1  08      FOR:    LXI     H,GETVAR ;LOOPS, ETC
1125    06B6    CD  66  00              CALL    PUSHJ   ;LOOK FOR "=" NEXT
1126    06B9    E5                      PUSH    H
1127    06BA    CD  05  02              CALL    SPNOR   ;IGNORE SPACES
1128    06BD    3A  85  14              LDA     CHAR
1129    06C0    FE  3D                  CPI     '='
1130    06C2    C4  DC  03              CNZ     ERROR4  ;LEFT OF = BAD, FOR OR SET
1131    06C5    CD  8D  00              CALL    GETC
1132    06C8    CD  B2  09              CALL    EVAL
1133    06CB    E1                      POP     H
1134    06CC    22  78  14              SHLD    PT1
1135    06CF    CD  D1  0F              CALL    TST
1136    06D2    2A  78  14              LHLD    PT1
1137    06D5    CD  B6  0F              CALL    STR
1138    06D8    97                      SUB     A
1139    06D9    01  20  15              LXI     B,TLIST
1140    06DC    21  D2  14              LXI     H,FLST1
1141    06DF    CD  E3  00              CALL    SORTJ   ;TEST LAST CHAR FROM EVAL
1142    06E2    CD  DC  03              CALL    ERROR4  ;EXCESS R-PAR
1143    06E5    2A  78  14      FINCR:  LHLD    PT1     ;SAVE VARIBLE ADDRESS
1144    06E8    E5                      PUSH    H
1145    06E9    21  AF  09              LXI     H,EVAL-3 ;EVALUATE INCREMENT IF ANY
1146    06EC    CD  66  00              CALL    PUSHJ
1147    06EF    97                      SUB     A
1148    06F0    01  20  15              LXI     B,TLIST
1149    06F3    21  CC  14              LXI     H,FLST2
1150    06F6    CD  E3  00              CALL    SORTJ   ;TEST TERMINATORS
1151    06F9    CD  DC  03              CALL    ERROR4  ;ILLEGAL TERMINATOR IN FOR
1152    06FC    CD  D1  0F      FLIMIT: CALL    TST
1153    06FF    21  4E  15              LXI     H,FLARG
1154    0702    CD  B6  0F              CALL    STR
1155    0705    21  4E  15              LXI     H,FLARG
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

1156     0708   CD 71 00                  CALL    PUSHF
1157     070B   21 AF 09                  LXI     H,EVAL-3  ;GET LIMIT (NO ERROR DETECTE
1158     070E   CD 66 00                  CALL    PUSHJ
1159     0711   CD D1 0F                  CALL    TST
1160     0714   21 4E 15                  LXI     H,FLARG
1161     0717   CD B6 0F                  CALL    STR
1162     071A   21 4E 15          FCONT:  LXI     H,FLARG   ;SAVE LIMIT
1163     071D   CD 71 00                  CALL    PUSHF
1164     0720   21 6F 14                  LXI     H,TEXTP
1165     0723   CD 71 00                  CALL    PUSHF     ;SAVE TEXT OF OBJECT STMTS
1166     0726   21 B4 05                  LXI     H,PROCESS
1167     0729   CD 66 00                  CALL    PUSHJ     ;DO OBJECT
1168     072C   21 6F 14                  LXI     H,TEXTP
1169     072F   CD 7F 00                  CALL    POPF      ;RESTORE REMAINING TEXT
1170     0732   21 4E 15                  LXI     H,FLARG
1171     0735   CD 7F 00                  CALL    POPF      ;GET LIMIT
1172     0738   21 78 15                  LXI     H,ITER1
1173     073B   CD 7F 00                  CALL    POPF      ;GET INC
1174     073E   E1                        POP     H
1175     073F   22 78 14                  SHLD    PT1       ;GET VARIBLE ADDRESS
1176     0742   CD E6 0F                  CALL    LOD
1177     0745   21 78 15                  LXI     H,ITER1
1178     0748   CD 4F 10                  CALL    AD
1179     074B   2A 78 14                  LHLD    PT1
1180     074E   CD B6 0F                  CALL    STR
1181     0751   21 78 15                  LXI     H,ITER1
1182     0754   CD E6 0F                  CALL    LOD
1183     0757   CD D1 0F                  CALL    TST
1184     075A   F5                        PUSH    PSW
1185     075B   2A 78 14                  LHLD    PT1
1186     075E   CD E6 0F                  CALL    LOD
1187     0761   F1                        POP     PSW
1188     0762   21 4E 15                  LXI     H,FLARG
1189     0765   F2 6F 07                  JP      OLDTST
1190     0768   CD 4C 10                  CALL    SB
1191     076B   F8                        RM
1192     076C   C3 76 07                  JMP     FORMR
1193     076F   CD 4C 10          OLDTST: CALL    SB
1194     0772   CA 76 07                  JZ      FORMR
1195     0775   F0                        RP
1196     0776   2A 78 14          FORMR:  LHLD    PT1
1197     0779   E5                        PUSH    H         ;SAVE ADDRESS
1198     077A   21 78 15                  LXI     H,ITER1
1199     077D   CD 71 00                  CALL    PUSHF     ;SAVE INC AGAIN
1200     0780   C3 1A 07                  JMP     FCONT
1201     0783   21 62 15          FINFIN: LXI     H,FLTONE
1202     0786   CD 71 00                  CALL    PUSHF     ;SET INC TO ONE
1203     0789   CD D1 0F                  CALL    TST
1204     078C   21 4E 15                  LXI     H,FLARG
1205     078F   CD B6 0F                  CALL    STR
1206     0792   C3 1A 07                  JMP     FCONT
1207
1208                              ; INPUT-OUTPUT STATEMENTS
1209     0795   97                ASK:    SUB     A         ;REMEMBER WHICH CALL
1210     0796   2F                        CMA
```

```
STMNT     LOC     OBJECT CODE       SOURCE STATEMENT

1211      0797    32  7E  14    TYPE:   STA     ATSW
1212      079A    97            TASK:   SUB     A
1213      079B    32  77  14            STA     DEBGSW      ;RE-ENABLE TRACE
1214      079E    01  1A  15            LXI     B,ALIST     ;SPECIAL CHAR?
1215      07A1    21  28  15            LXI     H,ATLIST
1216      07A4    CD  E3  00            CALL    SORTJ
1217      07A7    3A  7E  14            LDA     ATSW
1218      07AA    3C                    INR     A           ;TEST QUOTE SWITCH
1219      07AB    C2  FD  07            JNZ     TYPE2       ;TYPE
1220      07AE    21  B1  08            LXI     H,GETVAR
1221      07B1    CD  66  00            CALL    PUSHJ       ;DO ASK-SET UP PT1
1222      07B4    3A  85  14            LDA     CHAR        ;SAVE IN-LINE CHAR
1223      07B7    F5                    PUSH PSW
1224      07B8    E5                    PUSH    H
1225      07B9    3E  3A                MVI     A,':'       ;TYPE COLON
1226      07BB    CD  1B  01            CALL    PRNTC
1227      07BE    21  AB  15            LXI     H,IOBUF
1228      07C1    22  6D  14            SHLD    AXIN
1229      07C4    CD  51  01    AK2:    CALL    READC
1230      07C7    CD  BD  00            CALL    PACKC
1231      07CA    97                    SUB     A
1232      07CB    01  71  15            LXI     B,SPECIAL
1233      07CE    21  56  15            LXI     H,INFIX
1234      07D1    CD  E3  00            CALL    SORTJ
1235      07D4    C3  C4  07            JMP     AK2
1236      07D7    CD  BD  00    AK3:    CALL    PACKC
1237      07DA    2A  6F  14            LHLD    AXOUT
1238      07DD    E5                    PUSH    H
1239      07DE    21  AB  15            LXI     H,IOBUF
1240      07E1    22  6F  14            SHLD    AXOUT
1241      07E4    CD  AF  09            CALL    EVAL-3
1242      07E7    E1                    POP     H
1243      07E8    22  6F  14            SHLD    AXOUT
1244      07EB    CD  D1  0F            CALL    TST
1245      07EE    E1                    POP     H
1246      07EF    CD  B6  0F            CALL    STR
1247      07F2    F1            AK5:    POP     PSW
1248      07F3    32  85  14            STA     CHAR
1249      07F6    C3  95  07            JMP     ASK         ;CONTINUE
1250      07F9    E1            AK4:    POP     H
1251      07FA    C3  F2  07            JMP     AK5
1252      07FD    21  B2  09    TYPE2:  LXI     H,EVAL      ;DO TYPE
1253      0800    CD  66  00            CALL    PUSHJ
1254      0803    21  AB  15            LXI     H,IOBUF
1255      0806    CD  84  13            CALL    OU
1256      0809    21  AB  15            LXI     H,IOBUF
1257      080C    06  0D                MVI     B,13
1258      080E    7E            TYP:    MOV     A,M
1259      080F    C6  30                ADI     '0'
1260      0811    CD  1B  01            CALL    PRNTC
1261      0814    05                    DCR     B
1262      0815    23                    INX     H
1263      0816    C2  0E  08            JNZ     TYP
1264      0819    C3  97  07            JMP     TYPE
1265      081C    3E  01        TQUOT:  MVI     A,1         ;DISABLE TRACE
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

1266     081E   32 77 14                STA     DEBGSW
1267     0821   CD 8D 00        TQ2:    CALL    GETC        ;TYOE LITERALS
1268     0824   97                      SUB     A
1269     0825   01 7E 15                LXI     B,TLST2
1270     0828   21 52 15                LXI     H,TLST3
1271     082B   CD E3 00                CALL    SORTJ
1272     082E   97                      SUB     A
1273     082F   CD 1B 01                CALL    PRNTC
1274     0832   C3 21 08                JMP     TQ2
1275     0835   CD 8D 00        TINTR:  CALL    GETC        ;PASS %
1276     0838   CD 11 02                CALL    GETLN       ;READ FORMAT CONTROL
1277     083B   2A 86 14                LHLD    LINEND
1278     083E   22 70 15                SHLD    FISW        ;SAVE FORMAT CODE
1279     0841   C3 9A 07                JMP     TASK
1280     0844   3E 0D           TCRL2:  MVI     A,CR        ;SPLAR=CR ALONE
1281     0846   CD 1B 01                CALL    PRNTC
1282     0849   3E 0D           TCRLF:  MVI     A,CR        ;) IS BOTH
1283     084B   CD 45 01                CALL    XOUTL
1284     084E   CD 8D 00        TASK4:  CALL    GETC        ;MOVE TO NEXT CHARACTER
1285     0851   C3 9A 07                JMP     TASK
1286
1287                            ; SEARCH ROUTINES
1288     0854   CD 11 02        MODIFY: CALL    GETLN       ;READ LINENO
1289     0857   CD 9E 01                CALL    FINDLN      ;LOOP IT UP
1290     085A   CD DC 03                CALL    ERROR2      ;NON-EXISTENCE
1291     085D   CD 64 01                CALL    PRNTLN
1292     0860   2A 82 14                LHLD    BUFR        ;SET POINTERS
1293     0863   23                      INX     H
1294     0864   23                      INX     H
1295     0865   11 86 14                LXI     D,LINENO    ;COPY SAME LINE #
1296     0868   1A                      LDAX    D
1297     0869   77                      MOV     M,A
1298     086A   13                      INX     D
1299     086B   23                      INX     H
1300     086C   1A                      LDAX    D
1301     086D   77                      MOV     M,A
1302     086E   23                      INX     H
1303     086F   22 6D 14                SHLD    AXIN        ;FOR INPUT
1304     0872   CD 36 04        SCONT:  CALL    XI33        ;READ, NO ECHO
1305     0875   32 8E 14                STA     LIST3+1     ;SAVE SEARCH CHAR
1306     0878   3E 01                   MVI     A,1
1307     087A   32 77 14                STA     DEBGSW      ;NO BREAKS
1308     087D   CD 8D 00        SCHAR:  CALL    GETC        ;TYPE + TEST -F.F.
1309     0880   97                      SUB     A
1310     0881   CD 1B 01                CALL    PRNTC       ;TYPE
1311     0884   97                      SUB     A
1312     0885   01 8D 14                LXI     B,LIST3
1313     0888   21 16 15                LXI     H,LISTGO
1314     088B   CD E3 00                CALL    SORTJ       ;LOOK FOR MATCH
1315     088E   CD BD 00                CALL    PACKC       ;SAVE REW LINE
1316     0891   C3 7D 08                JMP     SCHAR
1317     0894   2A 82 14        SBAR:   LHLD    BUFR        ;RESTART BUFFER ADDRESS
1318     0897   23                      INX     H
1319     0898   23                      INX     H
1320     0899   23                      INX     H
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

1321    089A  23               INX   H
1322    089B  22 6D 14          SHLD  AXIN      ;SET POINTERS
1323    089E  CD 51 01  SFOUND: CALL  READC     ;READ KEYBOARD
1324    08A1  97               SUB   A
1325    08A2  01 88 14          LXI   B,LIST6
1326    08A5  21 0C 15          LXI   H,SRNLST
1327    08A8  CD E3 00          CALL  SORTJ
1328    08AB  CD BD 00  SGOT:   CALL  PACKC     ;PACK CHAR
1329    08AE  C3 9E 08          JMP   SFOUND    ;MORE
1330
1331                    ;GET A VARIABLE FROM VARIABLE LIST
1332                    ;EXIT- H & L POINT TO VALUE PART OF VARIABLE
1333                    ;PUSHES AXOUT PAST VARNAME
1334    08B1  CD 05 02  GETVAR: CALL  SPNOR     ;IGNORE BLANKS
1335    08B4  3A 85 14          LDA   CHAR      ;GET FIRST CHAR OF NAME
1336    08B7  47                MOV   B,A       ;SAVE
1337    08B8  C5                PUSH  B
1338    08B9  CD 8D 00          CALL  GETC      ;GET SCND CHAR OF NAME
1339    08BC  C1                POP   B
1340    08BD  CD F4 02          CALL  TESTC
1341    08C0  C3 D1 08          JMP   NOTA
1342    08C3  00                NOP
1343    08C4  00                NOP
1344    08C5  00                NOP
1345    08C6  00                NOP
1346    08C7  00                NOP
1347    08C8  00                NOP
1348    08C9  3A 85 14          LDA   CHAR
1349    08CC  4F                MOV   C,A       ;ALPHA--SAVE
1350    08CD  C5                PUSH  B
1351    08CE  C3 D7 08          JMP   GV2
1352    08D1  0E 40     NOTA:   MVI   C,'@'     ;SUBSTITUE FOR 2ND CHAR IF MISSING
1353    08D3  C5                PUSH  B
1354    08D4  C3 DA 08          JMP   GV2B
1355    08D7  CD 8D 00  GV2:    CALL  GETC
1356    08DA  CD F4 02  GV2B:   CALL  TESTC
1357    08DD  C3 E9 08          JMP   GV2A      ;T
1358    08E0  C3 D7 08          JMP   GV2       ;N
1359    08E3  C3 D7 08          JMP   GV2       ;F
1360    08E6  C3 D7 08          JMP   GV2       ;V
1361    08E9  CD D3 02  GV2A:   CALL  TSTLPR
1362    08EC  C3 32 09          JMP   NOSUB
1363    08EF  CD 80 09          CALL  ECALL
1364    08F2  21 35 00          LXI   H,ACCE+SCRB
1365    08F5  CD 71 00          CALL  PUSHF
1366    08F8  2A 78 14          LHLD  PT1
1367    08FB  CD E6 0F          CALL  LOD
1368    08FE  1E 20             MVI   E,32
1369    0900  CD 8E 12          CALL  FIX       ;CONVERT TO FIXED
1370    0903  5A                MOV   E,D       ;SAV SUBSCRIPT
1371    0904  51                MOV   D,C       ;LEAST SIGNIFICANT PORTION
1372    0905  EB                XCHG
1373    0906  22 7A 15          SHLD  TEMP
1374    0909  21 35 00          LXI   H,ACCE+SCRB
1375    090C  CD 7F 00          CALL  POPF
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

1376     090F   2A 7A 15                LHLD    TEMP
1377     0912   E5                      PUSH    H
1378     0913   2A 6F 14                LHLD    AXOUT     ;POINT POINTER PAST VARNAME
1379     0916   2B                      DCX     H
1380     0917   7E              GV5:    MOV     A,M
1381     0918   FE 29                   CPI     ')'
1382     091A   C4 DC 03                CNZ     ERROR4
1383     091D   CD 8D 00                CALL    GETC
1384     0920   CD F4 02        GV6A:   CALL    TESTC
1385     0923   C3 36 09                JMP     FNDVR-2
1386     0926   C3 2C 09                JMP     GV6       ;N
1387     0929   C3 2C 09                JMP     GV6       ;F
1388     092C   CD 8D 00        GV6:    CALL    GETC      ;V
1389     092F   C3 20 09                JMP     GV6A
1390     0932   11 00 00        NOSUB:  LXI     D,0       ;NO SUBSCRIPT MEANS MAKE IT ZERO
1391     0935   D5                      PUSH    D
1392     0936   D1                      POP     D
1393     0937   C1                      POP     B
1394     0938   2A 82 14        FNDVR:  LHLD    STRTV     ;DCHECK VARIABLE LIST
1395     093B   3A 7A 14        GV3:    LDA     LASTV     ;FIRST HALF OF
1396     093E   BD                      CMP     L         ;END OF VARS CHECK
1397     093F   CA 65 09                JZ      HC
1398     0942   7E              GV4:    MOV     A,M
1399     0943   B8                      CMP     B         ;CHECK VARNAME
1400     0944   23                      INX     H
1401     0945   C2 5B 09                JNZ     NOPE1
1402     0948   7E                      MOV     A,M
1403     0949   B9                      CMP     C         ;CHECK VARNAME 2ND  CHAR
1404     094A   23                      INX     H
1405     094B   C2 5C 09                JNZ     NOPE2
1406     094E   7E                      MOV     A,M
1407     094F   BA                      CMP     D         ;CHECK SUBSCRIPT
1408     0950   23                      INX     H
1409     0951   C2 5D 09                JNZ     NOPE3
1410     0954   7E                      MOV     A,M
1411     0955   BB                      CMP     E         ;CHECK 2ND HALF OF SUBSCRIPT
1412     0956   23                      INX     H
1413     0957   C8                      RZ                ;RETURN IF FOUND
1414     0958   C3 5E 09                JMP     NOPE
1415     095B   23              NOPE1:  INX     H
1416     095C   23              NOPE2:  INX     H
1417     095D   23              NOPE3:  INX     H
1418     095E   23              NOPE:   INX     H
1419     095F   23                      INX     H
1420     0960   23                      INX     H
1421     0961   23                      INX     H
1422     0962   C3 3B 09                JMP     GV3       ;TRY NEXT VARIABLE
1423     0965   3A 7B 14        HC:     LDA     LASTV+1   ;CHECK 2ND HALF END OF VARS
1424     0968   BC                      CMP     H
1425     0969   C2 42 09                JNZ     GV4       ;NOT END OF VARS
1426     096C   70                      MOV     M,B       ;NOT CREATED YET
1427     096D   23                      INX     H
1428     096E   71                      MOV     M,C
1429     096F   23                      INX     H
1430     0970   72                      MOV     M,D
```

```
STMNT     LOC   OBJECT CODE      SOURCE STATEMENT

1431      0971  23                      INX   H
1432      0972  73                      MOV   M,E
1433      0973  23                      INX   H           ;NOW CREATED
1434      0974  E5                      PUSH  H           ;SAVE POINTER TO VALUE
1435      0975  97                      SUB   A
1436      0976  77                      MOV   M,A
1437      0977  23                      INX   H
1438      0978  23                      INX   H
1439      0979  23                      INX   H
1440      097A  23                      INX   H
1441      097B  22  7A  14              SHLD  LASTV       ;MADE VARIABLE-SAVE NEW END OF LIST
1442      097E  E1                      POP   H
1443      097F  C9                      RET               ;RETURN POINTER TO VALUE
1444
1445      0980  21  35  00        ECALL:LXI   H,ACCE+SCRB
1446      0983  CD  71  00              CALL  PUSHF
1447      0986  3A  7D  14              LDA   LASTOP
1448      0989  F5                      PUSH  PSW
1449      098A  3A  81  14              LDA   STE
1450      098D  F5                      PUSH  PSW
1451      098E  21  AF  09              LXI   H,EVAL-3
1452      0991  CD  66  00              CALL  PUSHJ
1453      0994  F1                      POP   PSW
1454      0995  32  81  14              STA   STE
1455      0998  F1                      POP   PSW
1456      0999  32  7D  14              STA   LASTOP
1457      099C  CD  D1  0F              CALL  TST
1458      099F  21  6C  15              LXI   H,FLAC
1459      09A2  22  78  14              SHLD  PT1
1460      09A5  CD  B6  0F              CALL  STR
1461      09A8  21  35  00              LXI   H,ACCE+SCRB
1462      09AB  CD  7F  00              CALL  POPF
1463      09AE  C9                      RET
1464      09AF  CD  8D  00              CALL  GETC
1465      09B2  97              EVAL:   SUB   A
1466      09B3  32  81  14              STA   STE
1467      09B6  3A  85  14              LDA   CHAR
1468      09B9  FE  2B                  CPI   '+'         ;TEST FOR UNARY +
1469      09BB  CA  04  0A              JZ    ENUM
1470      09BE  FE  2D                  CPI   '-'         ;TEST FOR UNARY -
1471      09C0  CA  04  0A              JZ    ENUM
1472      09C3  CD  F4  02        EVALC:CALL  TESTC
1473      09C6  C3  ED  09              JMP   ETRM1       ;T
1474      09C9  C3  04  0A              JMP   ENUM        ;N
1475      09CC  C3  53  0A              JMP   EFUN        ;F
1476      09CF  CD  B1  08        VARGET:CALL GETVAR      ;VAR
1477      09D2  22  78  14              SHLD  PT1
1478      09D5  C3  98  0A              JMP   EDO
1479      09D8  3A  7C  14        OPNEXT:LDA  SRTCN
1480      09DB  32  7D  14              STA   LASTOP
1481      09DE  CD  F4  02              CALL  TESTC
1482      09E1  C3  F5  09              JMP   ETRMN       ;T
1483      09E4  CD  DC  03              CALL  ERROR4      ;N
1484      09E7  CD  DC  03              CALL  ERROR4      ;F
1485      09EA  CD  DC  03              CALL  ERROR4      ;VAR
```

```
STMNT     LOC     OBJECT CODE      SOURCE STATEMENT

1486      09ED    3A 7C 14         ETRM1:  LDA     SRTCN
1487      09F0    FE 06                    CPI     6          ;NULL PARENS
1488      09F2    CA D9 0A                 JZ      ELPAR
1489      09F5    CD D3 02         ETRMN:  CALL    TSTLPR
1490      09F8    C3 FE 09                 JMP     ETR2
1491      09FB    CD DC 03                 CALL    ERROR4     ;OP MISSING
1492      09FE    CD 8D 00         ETR2:   CALL    GETC
1493      0A01    C3 C3 09                 JMP     EVALC
1494      0A04    21 82 15         ENUM:   LXI     H,DATBUF
1495      0A07    E5                       PUSH    H
1496      0A08    01 3D 15         ENLP:   LXI     B,TERMS
1497      0A0B    CD FE 00                 CALL    SORTC
1498      0A0E    C3 2F 0A                 JMP     NUMD
1499      0A11    E1               ENL3:   POP     H
1500      0A12    3A 85 14                 LDA     CHAR
1501      0A15    D6 30                    SUI     600
1502      0A17    77                       MOV     M,A
1503      0A18    23                       INX     H
1504      0A19    FE 15                    CPI     'E'-600
1505      0A1B    C2 28 0A                 JNZ     ENL2
1506      0A1E    E5                       PUSH    H
1507      0A1F    CD 8D 00                 CALL    GETC
1508      0A22    CD 05 02                 CALL    SPNOR
1509      0A25    C3 11 0A                 JMP     ENL3
1510      0A28    E5               ENL2:   PUSH    H
1511      0A29    CD 8D 00                 CALL    GETC
1512      0A2C    C3 08 0A                 JMP     ENLP
1513      0A2F    E1               NUMD:   POP     H
1514      0A30    36 FF                    MVI     M,0FFH
1515      0A32    CD D1 0F                 CALL    TST
1516      0A35    21 4E 15                 LXI     H,FLARG
1517      0A38    CD B6 0F                 CALL    STR
1518      0A3B    21 82 15                 LXI     H,DATBUF
1519      0A3E    CD C2 12                 CALL    INP
1520      0A41    21 6C 15                 LXI     H,FLAC
1521      0A44    22 78 14                 SHLD    PT1
1522      0A47    CD B6 0F                 CALL    STR
1523      0A4A    21 4E 15                 LXI     H,FLARG
1524      0A4D    CD E6 0F                 CALL    LOD
1525      0A50    C3 98 0A                 JMP     EDO
1526      0A53    21 35 00         EFUN:   LXI     H,ACCE+SCRB
1527      0A56    CD 71 00                 CALL    PUSHF      ;SAVE ACCUM
1528      0A59    CD 8D 00                 CALL    GETC       ;GET FUNCTION CODE
1529      0A5C    CD 8D 00                 CALL    GETC
1530      0A5F    3A 85 14                 LDA     CHAR
1531      0A62    F5                       PUSH    PSW        ;SAVE FUNCTION CODE
1532      0A63    CD 8D 00         FUN3:   CALL    GETC       ;MOVE TO EXPRESSION
1533      0A66    CD F4 02                 CALL    TESTC
1534      0A69    C3 75 0A                 JMP     FUN2       ;T
1535      0A6C    C3 63 0A                 JMP     FUN3       ;N
1536      0A6F    C3 63 0A                 JMP     FUN3       ;F
1537      0A72    C3 63 0A                   JMP     FUN3     ;VAR
1538      0A75    CD D3 02         FUN2:   CALL    TSTLPR     ;MUST BE A (
1539      0A78    CD DC 03                 CALL    ERROR6     ;NO PAREN
1540      0A7B    CD 80 09         FUN4:   CALL    ECALL      ;EVALUATE FUNCTION INPUT
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

1541     0A7E   F1               POP     PSW         ;RESTORE FUNCTION CODE
1542     0A7F   21 8F 0A         LXI     H,EFUN3     ;GET RETURN ADDRESS
1543     0A82   E5               PUSH    H           ;SAVE ON STACK
1544     0A83   01 B7 14         LXI     B,FNTBL     ;GET FUNCTION CODES
1545     0A86   21 9F 14         LXI     H,FNTBF     ;ADDRESS OF FUNCTIONS
1546     0A89   CD E3 00         CALL    SORTJ       ;GOTO FUNCTION
1547     0A8C   CD DC 03         CALL    ERROR6      ;BAD FUNCTION CODE
1548     0A8F   21 35 00  EFUN3: LXI     H,ACCE+SCRB
1549     0A92   CD 7F 00         CALL    POPF        ;RETURN FROM FUNCTIONS, RESTORE ACCUM
1550     0A95   C3 EC 0A         JMP     ELPR2
1551                      ETERM:
1552     0A98   3A 81 14  EDO:   LDA     STE
1553     0A9B   A7               ANA     A
1554     0A9C   3E 01            MVI     A,1
1555     0A9E   32 81 14         STA     STE
1556     0AA1   C2 AD 0A         JNZ     EFST
1557     0AA4   21 E6 0F         LXI     H,LOD
1558     0AA7   22 C7 0A         SHLD    FLOP+1
1559     0AAA   C3 C3 0A         JMP     FLOP-3
1560     0AAD   3A 7D 14  EFST:  LDA     LASTOP
1561     0AB0   21 90 14         LXI     H,OPTBL
1562     0AB3   3D        ELP:   DCR     A
1563     0AB4   23               INX     H
1564     0AB5   23               INX     H
1565     0AB6   CA BC 0A         JZ      EFND
1566     0AB9   C3 B3 0A         JMP     ELP
1567     0ABC   5E        EFND:  MOV     E,M
1568     0ABD   23               INX     H
1569     0ABE   56               MOV     D,M
1570     0ABF   EB               XCHG
1571     0AC0   22 C7 0A         SHLD    FLOP+1
1572     0AC3   2A 78 14         LHLD    PT1
1573     0AC6   CD E6 0F  FLOP:  CALL    LOD
1574     0AC9   3A 33 00         LDA     OVER+SCRB
1575     0ACC   A7               ANA     A
1576     0ACD   C4 DC 03         CNZ     ERROR6      ;OVERFLOW
1577     0AD0   3A 7C 14         LDA     SRTCN
1578     0AD3   D6 09            SUI     9
1579     0AD5   FA D8 09         JM      OPNEXT
1580     0AD8   C9               RET
1581     0AD9   3A 7D 14  ELPAR: LDA     LASTOP
1582     0ADC   F5               PUSH    PSW
1583     0ADD   3A 81 14         LDA     STE
1584     0AE0   F5               PUSH    PSW
1585     0AE1   CD 80 09         CALL    ECALL
1586     0AE4   F1               POP     PSW
1587     0AE5   32 81 14         STA     STE
1588     0AE8   F1               POP     PSW
1589     0AE9   32 7D 14         STA     LASTOP
1590     0AEC   CD 8D 00  ELPR2: CALL    GETC
1591     0AEF   CD F4 02         CALL    TESTC       ;SET SRTCN
1592     0AF2   C3 98 0A         JMP     ETERM
1593     0AF5   C3 98 0A         JMP     ETERM
1594     0AF8   C3 98 0A         JMP     ETERM
1595     0AFB   C3 98 0A         JMP     ETERM
```

```
STMNT      LOC     OBJECT CODE       SOURCE STATEMENT

1596
1597                                 ; ERASE SINGLE LINES, GROUPS, OR VARIBLES
1598       0AFE    3E  01           ERASE:   MVI     A,1         ;TURN OFF TRACE
1599       0B00    32  77  14               STA     DEBGSW
1600       0B03    CD  F4  02               CALL    TESTC       ;TEST SECOND WORD IF ANY
1601       0B06    C3  60  0B               JMP     ERVX        ;ERASE VARIBLES
1602       0B09    C3  2E  0B               JMP     ERL         ;LINES OR GROUPS
1603       0B0C    CD  DC  03               CALL    ERROR3
1604       0B0F    3A  85  14               LDA     CHAR        ;ALL TEXT
1605       0B12    FE  41                   CPI     'A'
1606       0B14    C4  DC  03               CNZ     ERROR3      ;BAD ARG IN ERASE
1607       0B17    21  2A  16       ERT:    LXI     H,BUFBEG    ;ERASE ALL TEXT
1608       0B1A    22  82  14               SHLD    BUFR
1609       0B1D    21  25  16               LXI     H,CFRS
1610       0B20    36  00                   MVI     M,0
1611       0B22    23                       INX     H
1612       0B23    36  00                   MVI     M,0
1613       0B25    2A  82  14       ERV:    LHLD    STRTV       ;ERASE VARIBLES
1614       0B28    22  7A  14               SHLD    LASTV
1615       0B2B    C3  64  04               JMP     START       ;FIX POINTERS
1616       0B2E    CD  11  02       ERL:    CALL    GETLN       ;ERASE LINES
1617       0B31    2A  82  14               LHLD    BUFR        ;PROTECT REST OF LINE
1618       0B34    22  6D  14               SHLD    AXIN
1619       0B37    CD  29  03       ERG:    CALL    DELETE      ;EXTRACT A LINE
1620       0B3A    2A  73  14               LHLD    THISLN
1621       0B3D    23                       INX     H
1622       0B3E    23                       INX     H
1623       0B3F    22  73  14               SHLD    THISLN
1624       0B42    3A  84  14               LDA     NAGSW
1625       0B45    A7                       ANA     A
1626       0B46    FA  4B  0B               JM      ER1
1627       0B49    23                       INX     H
1628       0B4A    7E                       MOV     A,M
1629       0B4B    CD  E7  02       ER1:    CALL    TSTGRP      ;SKIP
1630       0B4E    C3  25  0B               JMP     ERV
1631       0B51    2A  73  14               LHLD    THISLN
1632       0B54    11  86  14               LXI     D,LINENO
1633       0B57    7E                       MOV     A,M
1634       0B58    12                       STAX    D
1635       0B59    23                       INX     H
1636       0B5A    13                       INX     D
1637       0B5B    7E                       MOV     A,M
1638       0B5C    12                       STAX    D
1639       0B5D    C3  37  0B               JMP     ERG
1640       0B60    2A  82  14       ERVX:   LHLD    STRTV       ;INIT VARIBLES BY
1641       0B63    22  7A  14               SHLD    LASTV       ;INDIRECT COMMAND
1642       0B66    C9                       RET
1643
1644                                 ; SYMBOL TABLE TYPEOUT ROUTINE
1645       0B67    3E  4E           TDUMP:  MVI     A,'N'
1646       0B69    CD  1B  01               CALL    PRNTC
1647       0B6C    3E  49                   MVI     A,'I'
1648       0B6E    CD  1B  01               CALL    PRNTC
1649       0B71    C9                       RET
1650
```

```
STMNT    LOC   OBJECT CODE      SOURCE STATEMENT

1651                           ;FLOATING POINT FUNCTIONS
1652                           ; INPUT TO FLOATING POINT FUNCTIONS IS POINTED TO BY PT1 (LHLD PT1)
1653                           ; INPUT IS 4 BYTES LONG.  IT USUALLY COMES FROM THE ARGUMENT TO THE FUNCTION.
1654                           ; OUTPUT FROM FLOATING POINT FUNCTIONS SHOULD BE THE FOUR BYTES POINTED
1655                           ; TO BY PT1.
1656   0B72   C9               XUSR:   RET
1657                           ;       INVERSE DIVIDE ROUTINE
1658   0B73   E5               IDV:    PUSH    H
1659   0B74   CD D1 0F                 CALL    TST         ;FLOATING POINT ACCUMULATOR TO REGISTERS
1660   0B77   2E 45                    MVI     L,>IDVT
1661   0B79   CD B6 0F                 CALL    STR         ;DIVISOR TO STOREAGE
1662   0B7C   E1                       POP     H
1663   0B7D   CD E6 0F                 CALL    LOD         ;DIVIDENE TO FLAOTING POINT ACCUMULATOR
1664   0B80   2E 45                    MVI     L,>IDVT     ;ADDRESS DIVISOR
1665   0B82   C3 2C 10                 JMP     DIV         ;RETURN THROUGH DIV ROUTINE
1666                           ;
1667                           ;       SUBROUTINE FOR EVALUATION OF ELEMENTARY
1668                           ;       FUNCTION MACLAURIN SERIES
1669                           ;
1670                           ;       ENTRY FMACE FOR EXPONENTIAL TYPE SERIES,E.G.
1671                           ;       ENTRY FMACL FOR LOGARITHMIC TYPE SERIES, E.G.
1672                           ;      ARCTAN(Z)=Z/1-Z**3/3+Z**5/5- ...
1673                           ;       S(I-1)=(1./A(I) + X*S(I)), S(N)=0.
1674                           ;
1675                           ;       IN BOTH SERIES DEL**2(A(I)) MUST BE CONSTANT
1676                           ;      ENTER WITH X IN FMACX, A(N) IN D, D(A(N-1)) IN C,
1677                           ;      D**2(A(1)) IN B.
1678                           ;      RESULTS IN FMACS, WHEN A(I) < 0.
1679                           ;
1680                           ;       2 LEVELS OF STACK USED BEYOND FLOATING POINT PACKAGE
1681                           ;
1682   0B85   AF               FMACL   XRA     A           ;CLEAR A-REGISTER FOR LOG TYPE SERIES
1683   0B86   21 49 00                 LXI     H,FMACS     ;POINT TO SIGMA
1684   0B89   77                       MOV     M,A         ;ZERO STORED
1685   0B8A   21 CB 0B                  LXI     H,FMACB    ;PRESET BRANCH B
1686   0B8D   C3 9F 0B                 JMP     FMACC       ;JOINT CODE
1687   0B90   2A F3 0B         FMACE:  LHLD    FONE        ;MOVE 1.0 TO SIGMA FOR EXP TYPE SERIES
1688   0B93   22 49 00                 SHLD    FMACS
1689   0B96   2A F5 0B                 LHLD    FONE+2
1690   0B99   22 4B 00                 SHLD    FMACS+2
1691   0B9C   21 BF 0B                 LXI     H,FMACA     ;PRESET BRANCH A
1692   0B9F   22 51 00         FMACC:  SHLD    FMACG       ;STORE PRESET BRANCH
1693   0BA2   1E 20                    MVI     E,32        ;COUNT FOR FLOATING OF A(I)
1694   0BA4   C5               FMACD:  PUSH    B           ;CHAIN RULE LOOP
1695   0BA5   D5                       PUSH    D           SAVE A(I), D(A(I)), D**2(A(1))
1696   0BA6   AF                       XRA     A           ;ZERO THE LEAD POSITIONS OF A(I)
1697   0BA7   47                       MOV     B,A
1698   0BA8   4F                       MOV     C,A
1699   0BA9   CD 77 12                 CALL    FLT         ;FLOAT A(I)
1700   0BAC   2E 4D                    MVI     L,>FMACT    ;INTO TEMP
1701   0BAE   CD B6 0F                 CALL    STR
1702   0BB1   2E 45                    MVI     L,>FMACX    ;LOAD THE ARGUMENT
1703   0BB3   CD E6 0F                 CALL    LOD
1704   0BB6   2E 49                    MVI     L,>FMACS    ;SIGMA SO FAR
1705   0BB8   CD 04 10                 CALL    MUL
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

1706    0BBB    2A  51  00              LHLD    FMACG       ;CHOOSE THE BRANCH
1707    0BBE    E9                      PCHL
1708    0BBF    21  4D  00      FMACA:  LXI     H,FMACT
1709    0BC2    CD  2C  10              CALL    DIV
1710    0BC5    21  F3  0B              LXI     H,FONE      ;POINTS TO 1.0
1711    0BC8    C3  DE  0B              JMP     FMACF       ;REJOIN COMMON CODE
1712    0BCB    21  49  00      FMACB:  LXI     H,FMACS
1713    0BCE    CD  B6  0F              CALL    STR         ;X*SIGMA
1714    0BD1    21  F3  0B              LXI     H,FONE      ;LOAD 1.0
1715    0BD4    CD  E6  0F              CALL    LOD
1716    0BD7    2E  4D                  MVI     L,>FMACT
1717    0BD9    CD  2C  10              CALL    DIV         ;1/A(I)
1718    0BDC    2E  49                  MVI     L,>FMACS    ;POINT TO X*SIGMA
1719    0BDE    CD  4F  10      FMACF:  CALL    AD
1720    0BE1    2E  49                  MVI     L,>FMACS    ;STORE NEXT LINK
1721    0BE3    CD  B6  0F              CALL    STR
1722    0BE6    D1                      POP     D           ;A(I) AND 32
1723    0BE7    C1                      POP     B           ;D(A) AND D**2(A)
1724    0BE8    7A                      MOV     A,D
1725    0BE9    91                      SUB     C
1726    0BEA    C8                      RZ                  ;DONE IF ZERO
1727    0BEB    D8                      RC                  ;OR NEGATIVE
1728    0BEC    57                      MOV     D,A         ;A(I-1)
1729    0BED    79                      MOV     A,C         ;D(A(I-1))
1730    0BEE    90                      SUB     B
1731    0BEF    4F                      MOV     C,A         ;D(A(I-2))
1732    0BF0    C3  A4  0B              JMP     FMACD       ;NEXT ITERATION
1733                            ; ???????????
1734    0BF3            81      FONE:   DC      81H
1735    0BF4            00              DC      0
1736    0BF5            00              DC      0
1737    0BF6            00              DC      0
1738    0BF7            81      FPIV2:  DC      81H
1739    0BF8            49              DC      49H
1740    0BF9            0F              DC      0FH
1741    0BFA            DC              DC      0DCH        ;PI/2
1742    0BFB            80      FLN2:   DC      80H
1743    0BFC            31              DC      31H
1744    0BFD            72              DC      72H
1745    0BFE            18              DC      18H         LN 2
1746                            ;
1747                            ;       SINE-COSINE USING MACLAURIN SERIES
1748                            ;
1749                            ;       ENTRY FSIN FOR SIN(X)
1750                            ;       ENTRY FCOS FOR COS(X)
1751                            ;             ENTER WITH X IN RADIANS IN FLOATING POINT ACCUMULATOR
1752                            ;       (IF ABS(X) > 2**24*PI, OVERFLOW FLAG IS SET)
1753                            ;
1754                            ;       WRITTEN BY O.C. JUELICH, 165-796, B6
1755                            ;       MISSILE SYSTEMS DIVISION, ROCKWELL INTERNATIONAL CORP.
1756                            ;       APRIL 1975
1757                            ;
1758                            ;       ENTRIES TO FLOATING POINT PACKAGE
1759                            ;
1760                            ;       3 LEVELS OF STACK USED BEYOND FLOATING POINT PACKAGE.
```

```
STMNT   LOC   OBJECT CODE     SOURCE STATEMENT

1761                         FCOS:
1762    08FF  2A 78 14               LHLD    PT1
1763    0C02  CD E6 0F               CALL    LOD
1764    0C05  CD C5 0F       FCOS0:  CALL    CHS         ;COMPLEMENT THE ANGLE
1765    0C08  21 F7 0B               LXI     H,FPIV2
1766    0C0B  CD 4F 10               CALL    AD
1767    0C0E  C3 17 0C               JMP     FSIN0
1768                         FSIN:
1769    0C11  2A 78 14               LHLD    PT1
1770    0C14  CD E6 0F               CALL    LOD
1771    0C17  CD D1 0F       FSIN0:  CALL    TST         ;FETCH ARGUMENT
1772    0C1A  2E 55                  MVI     L,>FSINX    ;TO SAVE IT
1773    0C1C  CD B6 0F               CALL    STR
1774    0C1F  21 F7 0B               LXI     H,FPIV2     ;REDUCE X TO REVOLUTIONS * 4
1775    0C22  CD 2C 10               CALL    DIV
1776    0C25  1E 1A                  MVI     E,26
1777    0C27  CD 8E 12               CALL    FIX
1778    0C2A  DA 42 10               JC      OVERF       ;QUIT IF ANGLE TOO LOARGE
1779    0C2D  1E 1A                  MVI     E,26
1780    0C2F  16 00                  MVI     D,0         ;WIPE OUT FRACTIONAL REVOLUTIONS
1781    0C31  CD 77 12               CALL    FLT         ;INTEGER PART OF REVOLUTIONS
1782    0C34  21 F7 0B               LXI     H,FPIV2      ;TO RADIANS
1783    0C37  CD 04 10               CALL    MUL
1784    0C3A  CD C5 0F               CALL    CHS         ;SUBTRACT INTEGRAL PART
1785    0C3D  2E 55          FSINA:  MVI     L,>FSINX    ;SUM IS REDUCED
1786    0C3F  CD 4F 10               CALL    AD
1787    0C42  2E 55                  MVI     L,>FSINX    ;SAVE IT
1788    0C44  CD B6 0F               CALL    STR
1789    0C47  CD C8 0F               CALL    ABS         ;FORCE ANGLE INTO REDUCED RANGE
1790    0C4A  21 F7 0B               LXI     H,FPIV2
1791    0C4D  CD 4C 10               CALL    SB
1792    0C50  FA 71 0C               JM      FSINB       ;IF NEGATIVE OR ZERO
1793    0C53  CA 71 0C               JZ      FSINB       ;THEN ANGLE IS REDUCED
1794    0C56  21 F7 0B               LXI     H,FPIV2      ;ABS(X)-PI
1795    0C59  CD 4C 10               CALL    SB
1796    0C5C  5F                     MOV     E,A         ;SAVE A-REGISTER
1797    0C5D  2E 56                  MVI     L,>FSINX+1
1798    0C5F  7E                     MOV     A,M
1799    0C60  E6 80                  ANI     80H         ;SIGN OF X
1800    0C62  EE 80                  XRI     80H         ;INVERTED
1801    0C64  A8                     XRA     B           ;=SIGN(X)*(ABS(X)-PI)
1802    0C65  47                     MOV     B,A
1803    0C66  7B                     MOV     A,E         ;RESTORE A-REGISTER
1804    0C67  2B                     DCX     H           ;POINT TO FSINX
1805    0C68  CD B6 0F               CALL    STR         ;REDUCED X
1806    0C6B  CD BE 0F               CALL    ZRO         ;CLEAR ACCUMULATOR
1807    0C6E  C3 3D 0C               JMP     FSINA       ;REPEAT UNTIL ABS(X) <= PI/2
1808                         ;
1809    0C71  2E 55          FSINB:  MVI     L,>FSINX
1810    0C73  CD E6 0F               CALL    LOD
1811    0C76  2E 55                  MVI     L,>FSINX
1812    0C78  CD 04 10               CALL    MUL
1813    0C7B  CD C5 0F               CALL    CHS         ;-X**2
1814    0C7E  2E 45                  MVI     L,>FMACX
1815    0C80  CD B6 0F               CALL    STR         ;TO MACLAURIN SERIES
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

1816     0C83   16 48                   MVI   D,72      ;9*8, 11 TERM DISCARDED, 18 BITS PRECISION
1817     0C85   0E 1E                   MVI   C,30      ;9*8-7*6
1818     0C87   06 08                   MVI   B,8       ;(9*8-7*6)-(7*6-5*4)
1819     0C89   CD 90 0B                CALL  FMACE
1820     0C8C   2E 49                   MVI   L,>FMACS  ;SUM OF SERIES / X
1821     0C8E   CD E6 0F                CALL  LOD
1822     0C91   2E 55                   MVI   L,>FSINX
1823     0C93   CD 04 10                CALL  MUL
1824     0C96   FE 81                   CPI   81H       ;SEE IF TAIL NEEDS CLEANING
1825     0C98   DA A1 0C                JC    FSINC     ;NO, MAGNITUDE IS < 1./
1826     0C9B   2E 38                   MVI   L,ACC2
1827     0C9D   AF                      XRA   A         ;ZEROS FOR THE TAIL
1828     0C9E   77                      MOV   M,A
1829     0C9F   2C                      INR   L
1830     0CA0   77                      MOV   M,A
1831     0CA1   CD D1 0F        FSINC:  CALL  TST       ;RESTORE FLAGS AND REGISTERS
1832     0CA4   2A 78 14                LHLD  PT1
1833     0CA7   CD B6 0F                CALL  STR
1834     0CAA   C9                      RET
1835                            ;       ARTAN ROUTINE USING MACLAURIN SERIES
1836                            ;       ENTRY FATAN FOR ARCTAN(X), WITH X IN FLOATING ACCUMULATOR
1837                            ;       RESULT RETURNED IN FLOATING ACCUMULATOR
1838                            ;       WRITTEN BY O.C. JUELICH
1839                            ;
1840                            ;       FOUR LEVELS OF STACK USED BEYOND FLOATING POINT PACKAGE
1841                            ARTN:
1842     0CAB   2A 78 14                LHLD  PT1
1843     0CAE   CD E6 0F                CALL  LOD
1844     0CB1   CD D1 0F        FATAN:  CALL  TST       ;GET F. P. ACC. INTO REGISTERS
1845     0CB4   C8                      RZ
1846     0CB5   FE 81                   CPI   81H       ;TEST EXPONENT
1847     0CB7   DA ED 0C                JC    FATN1     ;RETURN TO CALLER FROM FATN1
1848     0CBA   21 F3 0B                LXI   H,FONE    ;1.0
1849     0CBD   CD 73 0B                CALL  IDV       ;1.0/X
1850     0CC0   CD ED 0C                CALL  FATN1     ;GET ARCTAN(1/X)
1851     0CC3   2E 59                   MVI   L,>FATNU  ;SIGN(T)*(PI/2-ABS(T))
1852     0CC5   CD B6 0F                CALL  STR
1853     0CC8   21 F7 0B                LXI   H,FPIV2   ;PI/2
1854     0CCB   CD E6 0F                CALL  LOD
1855     0CCE   5F                      MOV   E,A       ;SAVE A-REGISTER
1856     0CCF   2E 5A                   MVI   L,>FATNU+1
1857     0CD1   7E                      MOV   A,M       ;TO A-REGISTER
1858     0CD2   E6 80                   ANI   80H
1859     0CD4   B0                      ORA   B         ;ATTACH TO PI/2
1860     0CD5   47                      MOV   B,A
1861     0CD6   7B                      MOV   A,E       ;RESTORE A-REGISTER
1862     0CD7   2E 55                   MVI   L,>FATNT  ;SAVE SIGN(T)*PI/2
1863     0CD9   CD B6 0F                CALL  STR
1864     0CDC   2E 55                   MVI   L,>FATNT
1865     0CDE   CD E6 0F                CALL  LOD
1866     0CE1   2E 59                   MVI   L,>FATNU  ;=SIGN (T)*(PI/2-ABS(T))
1867     0CE3   CD 4C 10                CALL  SB        ;=SIGN(T)*ABS(T)=T
1868     0CE6   2A 78 14                LHLD  PT1
1869     0CE9   CD B6 0F                CALL  STR
1870     0CEC   C9                      RET
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

1871                          ;       EVALUATE ARCTAN OF ARGUEMNTS & 1.
1872    0CED  2E 55           FATN1:  MVI     L,>FATNT  ;POINT TO TEMP
1873    0CEF  CD B6 0F                CALL    STR
1874    0CF2  2E 55                   MVI     L,>FATNT
1875    0CF4  CD 04 10                CALL    MUL       ;TAN(T)**2
1876    0CF7  21 F3 0B                LXI     H,FONE
1877    0CFA  CD 4F 10                CALL    AD
1878    0CFD  CD 9E 0E                CALL    FSQRT
1879    0D00  21 F3 0B                LXI     H,FONE
1880    0D03  CD 4F 10                CALL    AD        ;1.0+SQRT(TAN(T)**2+1.0)
1881    0D06  2E 55                   MVI     L,>FATNT  ;TAN(T)
1882    0D08  CD 73 0B                CALL    IDV       ;TAN(T/2)
1883    0D0B  2E 55                   MVI     L,>FATNT
1884    0D0D  CD B6 0F                CALL    STR
1885    0D10  2E 59                   MVI     L,>FATNU
1886    0D12  3C                      INR     A         ;2*TAN(T/2)
1887    0D13  CD B6 0F                CALL    STR
1888    0D16  2E 55                   MVI     L,>FATNT
1889    0D18  CD 04 10                CALL    MUL
1890    0D1B  CD C5 0F                CALL    CHS       ;-TAN(T/2)**2
1891    0D1E  2E 45                   MVI     L,>FMACX
1892    0D20  CD B6 0F                CALL    STR
1893    0D23  16 0B                   MVI     D,11      ;TERM 13 DISCARDED, 16 BITS PRECISION IN RANGE
1894    0D25  0E 02                   MVI     C,2       ;(11-9)
1895    0D27  06 00                   MVI     B,0       ;(11-9)-(9-7)
1896    0D29  CD 85 0B                CALL    FMACL
1897    0D2C  2E 49                   MVI     L,>FMACS  ;(T/2)/TAN(T/2)
1898    0D2E  CD E6 0F                CALL    LOD
1899    0D31  2E 59                   MVI     L,>FATNU  ;*(2*TAN(T/2))
1900    0D33  CD 04 10                CALL    MUL
1901    0D36  E5                      PUSH    H
1902    0D37  2A 78 14                LHLD    PT1
1903    0D3A  CD B6 0F                CALL    STR
1904    0D3D  E1                      POP     H
1905    0D3E  C9                      RET
1906                          ;       EXPONENTIAL AND HYBPERBOLIC SIN ROUTINE
1907                          ;       USING MACLAURIN SERIES FOR SINH
1908                          ;       ENTRY FEXP FOR EXP(X)
1909                          ;       ENTRY FSINH FOR FSINH(X)
1910                          ;               ENTER WITH X IN FLOATING POINT ACCUMULATOR
1911                          ;               RETURNS WITH FUNCTION IN FOOATING POINT ACCUMULATOR
1912                          ;               IF FUNCTION EXCEEDS 2**127, OVERFLOW FLAG IS SET
1913                          ;       WRITTEN BY O.C. JUELICH
1914                          ;       5 LEVELS OF STACK USED BEYOND FLOATING POINT PACKAGE
1915                          FHYS:
1916    0D3F  2A 78 14                LHLD    PT1
1917    0D42  CD E6 0F                CALL    LOD
1918    0D45  CD D1 0F        FSINH:  CALL    TST       ;FETCH FLOATING POINT ACCUMULATOR
1919    0D48  2E 55                   MVI     L,>FSNHX  ;SAVE ARGUMENT
1920    0D4A  CD B6 0F                CALL    STR
1921    0D4D  2E 53                   MVI     L,>FSNHD  ;ADDRESS DOUBLING COUNTER
1922    0D4F  36 00                   MVI     M,0
1923    0D51  D6 80                   SUI     80H       ;REMOVE OFFSET FROM A
1924    0D53  FA 63 0D                JM      FSNHA     ;DOUBLING COUNT AND X ARE O.K.
1925    0D56  FE 08                   CPI     8         ;ELIMINATE OVERSIZE DOUNBLING
```

```
STMNT   LOC   OBJECT CODE     SOURCE STATEMENT

1926    0D58  F2 42 10              JP      OVERF     ;RETURN THROUGH OVERFLOW ROUTINE
1927    0D5B  77                    MOV     M,A       ;SAVE THE DOUBLING ARGUMENTS
1928    0D5C  2E 55                 MVI     L,>FSNHX  ;BRING ARGUMENT INTP RANGE
1929    0D5E  36 80                 MVI     M,80H
1930    0D60  CD E6 0F              CALL    LOD       ;PUT X INTO FLOATING ACCUMULATOR
1931    0D63  2E 55           FSNHA: MVI    L,>FSNHX
1932    0D65  CD 04 10              CALL    MUL       ;X**2
1933    0D68  2E 45                 MVI     L,>FMACX
1934    0D6A  CD B6 0F              CALL    STR
1935    0D6D  16 2A                 MVI     D,42      ;7*6, 9 TERM DISCARDED, 18 BITS PRECISION
1936    0D6F  0E 16                 MVI     C,22      ;7*6-5*4
1937    0D71  06 08                 MVI     P 8       ;(7*6-5*4)-(5*4-3*2)
1938    0D73  CD 90 0B              CALL    FMACE
1939    0D76  2E 49                 MVI     L,>FMACS
1940    0D78  CD E6 0F              CALL    LOD
1941    0D7B  2E 55                 MVI     L,>FSNHX
1942    0D7D  CD 04 10              CALL    MUL
1943    0D80  2E 55                 MVI     L,>FSNHX  ;SINH(X)
1944    0D82  CD B6 0F              CALL    STR
1945    0D85  2E 55                 MVI     L,>FSNHX  ;SINH(X)**2
1946    0D87  CD 04 10              CALL    MUL
1947    0D8A  21 F3 0B              LXI     H,FONE    ;+1.0
1948    0D8D  CD 4F 10              CALL    AD
1949    0D90  CD 9E 0E              CALL    FSQRT     ;COSH(X) FOR DOUBLING AND FOR EXP(X-
1950    0D93  2E 45                 MVI     L,>FMACX  ;TEMP
1951    0D95  CD B6 0F              CALL    STR
1952    0D98  2E 53           FSNHB: MVI    L,>FSNHD   ;ADDRESS DOUBLING COUNT
1953    0D9A  35                    DCR     M         ;TALLY AT LOOP TOP
1954    0D9B  FA C9 0D              JM      FSNHC     ;DONE WHEN NEGATIVE
1955    0D9E  2E 45                 MVI     L,>FMACX  ;COSH(X/2)
1956    0DA0  CD E6 0F              CALL    LOD
1957    0DA3  2E 55                 MVI     L,>FSNHX  ;SINH(X/2)
1958    0DA5  CD 04 10              CALL    MUL
1959    0DA8  3C                    INR     A         ;2.*SINH(X/2)*COSH(X/2)
1960    0DA9  2E 55                 MVI     L,>FSNHX  ;SINH(X)
1961    0DAB  CD B6 0F              CALL    STR
1962    0DAE  2E 45                 MVI     L,>FMACX  ;COSH(X/2)
1963    0DB0  CD E6 0F              CALL    LOD
1964    0DB3  2E 45                 MVI     L,>FMACX
1965    0DB5  CD 04 10              CALL    MUL
1966    0DB8  2E 35                 MVI     L,ACCE    ;2.*COSH(X/2)**2
1967    0DBA  34                    INR     M
1968    0DBB  21 F3 0B              LXI     H,FONE    ;-1.
1969    0DBE  CD 4C 10              CALL    SB
1970    0DC1  2E 45                 MVI     L,>FMACX  ;=COSH(X)
1971    0DC3  CD B6 0F              CALL    STR
1972    0DC6  C3 98 0D              JMP     FSNHB     ;TEST THE DOUBLING COUNT
1973    0DC9  2E 55           FSNHC: MVI    L,>FSNHX
1974    0DCB  CD E6 0F              CALL    LOD
1975    0DCE  E5                    PUSH    H
1976    0DCF  2A 78 14              LHLD    PT1
1977    0DD2  CD B6 0F              CALL    STR
1978    0DD5  E1                    POP     H
1979    0DD6  C9                    RET
1980                  FEXP:
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

1981    0DD7    2A 78 14                 LHLD    PT1
1982    0DDA    CD E6 0F                 CALL    LOD
1983    0DDD    CD D1 0F        FEXP0:   CALL    TST
1984    0DE0    F2 09 0E                 JP      FEXPP
1985    0DE3    2E 33                    MVI     L,OVER     ;SAVE OVERFLOW FLAG
1986    0DE5    5E                       MOV     E,M
1987    0DE6    36 00                    MVI     M,0        ;CLEAR OVERFLOW FLAG
1988    0DE8    2E 4F                    MVI     L,FEXOV-SCR
1989    0DEA    73                       MOV     M,E        ;OLD FLAG TO SAVE CELL
1990    0DEB    CD C8 0F                 CALL    ABS
1991    0DEE    CD 09 0E                 CALL    FEXPP      ;EXP(-X) IN ACC
1992    0DF1    2E 54                    MVI     L,>FEXOV   ;GET OLD OVERFLOW FLAG BACK
1993    0DF3    5E                       MOV     E,M
1994    0DF4    2E 33                    MVI     L,>OVER    ;PICK UP NEW ONE TO TEST
1995    0DF6    7E                       MOV     A,M
1996    0DF7    73                       MOV     M,E        ;RESTORE OLD OVERFLOW FLAG
1997    0DF8    A7                       ANA     A          ;SET FLAGS
1998    0DF9    C2 BE 0F                 JNZ     ZRO        ;RECIPROCAL OF OVERFLOW IS ZERO
1999    0DFC    21 F3 0B                 LXI     H,FONE
2000    0DFF    CD 73 0B                 CALL    IDV        (1./EXP(-X)=EXP(X)
2001    0E02    2A 78 14                 LHLD    PT1
2002    0E05    CD B6 0F                 CALL    STR
2003    0E08    C9                       RET
2004    0E09    CD 45 0D        FEXPP:   CALL    FSINH      ;SINH(X)
2005    0E0C    2E 45                    MVI     L,>FMACX   ;+COSH(X)
2006    0E0E    CD 4F 10                 CALL    AD         ;=EXP(X)
2007    0E11    E5                       PUSH    H
2008    0E12    2A 78 14                 LHLD    PT1
2009    0E15    CD B6 0F                 CALL    STR
2010    0E18    E1                       POP     H
2011    0E19    C9                       RET
2012                            ; NATURAL LOGARITHM ROUTINE USING MACLAURIN SERIES
2013                            ; ENTRY FLOG FOR LN(ANS(X)), WITH X IN FLOATING POINT ACCUMULATOR
2014                            ;       RESULT IS RETURNED IN FLOATING POINT ACCUMULATOR
2015                            ;       IF X = 0 THE OVERFLOW FLAG IS SET
2016                            ; 3 LEVELS OF STACK USED
2017                            FLOG:
2018    0E1A    2A 78 14                 LHLD    PT1
2019    0E1D    CD E6 0F                 CALL    LOD
2020    0E20    CD C8 0F        FLOG0:   CALL    ABS        ;FORCE ARGUMENT POSITIVE, SET ZERO FLAG
2021    0E23    CA 42 10                 JZ      OVERF      ;RETURN THROUGH OVERFLOW ROUTINE
2022    0E26    D6 81                    SUI     81H        ;REMOVE EXPONENT OFFSET
2023    0E28    2E 53                    MVI     L,>FLOGE
2024    0E2A    77                       MOV     M,A
2025    0E2B    3E 81                    MVI     A,81H      ;NORMALIZE ARGUEMENT
2026    0E2D    2E 56                    MVI     L,>FLOGX
2027    0E2F    CD B6 0F                 CALL    STR        ;CALL IT X
2028    0E32    2E 56                    MVI     L,>FLOGX
2029    0E34    CD E6 0F                 CALL    LOD
2030    0E37    21 F3 0B                 LXI     H,FONE
2031    0E3A    CD 4F 10                 CALL    AD
2032    0E3D    2E 49                    MVI     L,>FMACS
2033    0E3F    CD B6 0F                 CALL    STR        ;X+1.0
2034    0E42    2E 56                    MVI     L,>FLOGX
2035    0E44    CD E6 0F                 CALL    LOD
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

2036   0E47   21 F3 0B            LXI   H,FONE
2037   0E4A   CD 4C 10            CALL  SB          ;X-1./
2038   0E4D   2E 49               MVI   L,>FMACS
2039   0E4F   CD 2C 10            CALL  DIV
2040   0E52   2E 5B               MVI   L,FLOGX
2041   0E54   CD B6 0F            CALL  STR         (X-1.0)/(X+1.0)
2042   0E57   2E 50               MVI   L,>FLOGX
2043   0E59   CD 04 10            CALL  MUL
2044   0E5C   2E 45               MVI   L,>FMACX
2045   0E5E   CD B6 0F            CALL  STR         ;((X-1.0)/(X+1.0))*2
2046   0E61   16 09               MVI   D,9         ;DISCARD 11 TERM FOR 18 BIT PRECISION
2047   0E63   0E 02               MVI   C,2         ;9-7
2048   0E65   06 00               MVI   B,0         ;(9-7)-(7-5)
2049   0E67   CD 85 0B            CALL  FMACL
2050   0E6A   2E 49               MVI   L,>FMACS
2051   0E6C   34                  INR   M           ;DOUBLE THE SUM
2052   0E6D   CD E6 0F            CALL  LOD
2053   0E70   2E 5B               MVI   L,FLOGX
2054   0E72   CD 04 10            CALL  MUL         ;LOGARITHM OF FRACTIONAL PART
2055   0E75   2E 5B               MVI   L,>FLOGX
2056   0E77   CD B6 0F            CALL  STR
2057   0E7A   2E 53               MVI   L,>FLOGE
2058   0E7C   7E                  MOV   A,M         ;EXPONENT AS INTEGER
2059   0E7D   06 00               MVI   B,0
2060   0E7F   48                  MOV   C,B
2061   0E80   50                  MOV   D,B
2062   0E81   1E 08               MVI   E,8         ;BINARY SCALE FOR EXPONENT
2063   0E83   CD 77 12            CALL  FLT
2064   0E86   21 FB 0B            LXI   H,FLN2
2065   0E89   CD 04 10            CALL  MUL         ;LOGARITHM OF 2**EXPONENT
2066   0E8C   2E 5B               MVI   L,>FLOGX    ;LOGARITHM OF FRACTIONAL PART
2067   0E8E   CD 4F 10            CALL  AD
2068   0E91   2A 78 14            LHLD  PT1
2069   0E94   CD B6 0F            CALL  STR
2070   0E97   C9                  RET
2071   0E98   2A 78 14   XSQT:     LHLD  PT1         ;POOINT TO INPUT
2072   0E9B   CD E6 0F            CALL  LOD         ;LOAD IT
2073                     ;     ONE LEVEL OF STACK USED, BEYOND FLOATING PACKAGE
2074   0E9E   CD C8 0F   FSQRT:    CALL  ABS         ;FORCE ARGUMNET PSOITIVE, SET ZERRO FLAG
2075   0EA1   C8                  RZ                ;RETURN ON ZERO
2076   0EA2   2E 45               MVI   L,>FSQRN    ;STRRAGE FOR ARGUMENY
2077   0EA4   CD B6 0F            CALL  STR
2078   0EA7   A7                  ANA   A           ;RESET CARRY BIT
2079   0EA8   1F                  RAR               ;HALVE THE EXPONENT
2080   0EA9   C6 40               ADI   40H         ;RESTORE THE OFFSET
2081   0EAB   2E 49               MVI   L,>FSQRX    ;STORE THE FIRST
2082                     ;     ITERATE
2083   0EAD   CD B6 0F            CALL  STR
2084   0EB0   16 05               MVI   D,5         ;ITERATION COUNT
2085   0EB2   D5                  PUSH  D           ;STACKED
2086   0EB3   2E 45      FSQRL:   MVI   L,>FSQRN    ;LOAD THE ARGUMENT
2087   0EB5   CD E6 0F            CALL  LOD
2088   0EB8   2E 49               MVI   L,>FSQRX    ;DIVIDE BY ITERATE
2089   0EBA   CD 2C 10            CALL  DIV
2090   0EBD   2E 49               MVI   L,>FSQRX    ;ADD ITERATE
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

2091    0EBF    CD 4F 10                 CALL    AD
2092    0EC2    D6 01                    SUI     1          ;VHALVE THE RESULT
2093    0EC4    2E 49                    MVI     L,>FSQRX   ;RESTORE NEXT ITERATE
2094    0EC6    CD B6 0F                 CALL    STR
2095    0EC9    D1                       POP     D          ;RESTORE ITERATION COUNT
2096    0ECA    15                       DCR     D          ;TALLY
2097    0ECB    CA D2 0E                 JZ      FSQRE      ;EXIT WHEN COUNT EXHAUSTED
2098    0ECE    D5                       PUSH    D          ;;SAVE IT OTHERWISE
2099    0ECF    C3 B3 0E                 JMP     FSQRL      ;TO NEXT ITERATION
2100    0ED2    2E 49            FSQRE:  MVI     L,>FSQRX   ;RESULT TO ACCUMULATOR
2101    0ED4    CD E6 0F                 CALL    LOD
2102    0ED7    E5                       PUSH    H
2103    0ED8    2A 78 14                 LHLD    PT1        ;POINT TO OUTPUT
2104    0EDB    CD B6 0F                 CALL    STR
2105    0EDE    E1                       POP     H
2106    0EDF    C9                       RET
2107
2108                             ; PSUEDO RANDOM NUMBER GENERATOR
2109    0EE0    21 27 15         XRAN:   LXI     H,RAN0+3
2110    0EE3    06 08                    MVI     B,8
2111    0EE5    7E                       MOV     A,M
2112    0EE6    07               RN1:    RLC
2113    0EE7    07                       RLC
2114    0EE8    07                       RLC
2115    0EE9    AE                       XRA     M
2116    0EEA    17                       RAL
2117    0EEB    17                       RAL
2118    0EEC    2B                       DCX     H
2119    0EED    2B                       DCX     H
2120    0EEE    2B                       DCX     H
2121    0EEF    0E 04                    MVI     C,4
2122    0EF1    7E               RN2:    MOV     A,M
2123    0EF2    17                       RAL
2124    0EF3    77                       MOV     M,A
2125    0EF4    23                       INX     H
2126    0EF5    0D                       DCR     C
2127    0EF6    C2 F1 0E                 JNZ     RN2
2128    0EF9    2B                       DCX     H
2129    0EFA    05                       DCR     B
2130    0EFB    C2 E6 0E                 JNZ     RN1
2131    0EFE    21 24 15                 LXI     H,RAN0
2132    0F01    CD E6 0F                 CALL    LOD
2133    0F04    2A 78 14                 LHLD    PT1
2134    0F07    3E 80                    MVI     A,80H
2135    0F09    CD B6 0F                 CALL    STR
2136    0F0C    C9                       RET
2137    0F0D    2A 78 14         XABS:   LHLD    PT1        ;GET VALUE TO OPERATE ON
2138    0F10    CD E6 0F                 CALL    LOD        ;LOAD IT
2139    0F13    CD C8 0F                 CALL    ABS        ;TAKE ABS
2140    0F16    2A 78 14                 LHLD    PT1        ;GET ADDRESS TO PUT IT
2141    0F19    CD B6 0F                 CALL    STR        ;SAVE RESULT
2142    0F1C    C9                       RET
2143    0F1D    2A 78 14         XINT:   LHLD    PT1        ;GET VALUE TO OPERATE ON
2144    0F20    CD E6 0F                 CALL    LOD        ;LOAD IT
2145    0F23    1E 20                    MVI     E,32       ;SCALING FACTOR IS 32
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

2146     0F25   CD 8E 12                CALL   FIX        ;TAKE INTEGER
2147     0F28   1E 20                   MVI    E,32
2148     0F2A   CD 77 12                CALL   FLT        ;FLOAT RESULT
2149     0F2D   2A 78 14                LHLD   PT1        ;GET RESTORE ADDRESS
2150     0F30   CD B6 0F                CALL   STR        ;SAVE RESULT
2151     0F33   C9                      RET
2152     0F34   2A 78 14        XSGN:   LHLD   PT1        ;GET ADDRESS OF OPERAND
2153     0F37   CD E6 0F                CALL   LOD        ;LOAD IT
2154     0F3A   CD D1 0F                CALL   TST        ;TEST IFRT
2155     0F3D   CA 50 0F                JZ     XSG1       ;ZERO
2156     0F40   FA 54 0F                JM     XSG2       ;LESS THAN ZERO
2157     0F43   21 62 15                LXI    H,FLTONE   ;>0, RESULT=1
2158     0F46   CD E6 0F                CALL   LOD        ;LOAD 1
2159     0F49   2A 78 14        XSG3:   LHLD   PT1        ;GET POINTER TO RESULT
2160     0F4C   CD B6 0F                CALL   STR        ;STORE RESULT
2161     0F4F   C9                      RET
2162     0F50   97              XSG1:   SUB    A          ;MAKE EXPONENT ZERO
2163     0F51   C3 49 0F                JMP    XSG3       ;RESULT IS ZERO
2164     0F54   21 62 15        XSG2:   LXI    H,FLTONE   ;<0
2165     0F57   CD E6 0F                CALL   LOD        ;GET ONE
2166     0F5A   CD C5 0F                CALL   CHS        ;MAKE IT NEG
2167     0F5D   C3 49 0F                JMP    XSG3
2168                             ;
2169                             ;RAISE TO A POWER
2170                             ;ARGUMENT MUST BE POSITIVE INTEGER
2171                             ;(WORKS LIKE A FLOATING POINT PACKAGE ROUTINE)
2172     0F60   21 35 00        FLEX:   LXI    H,ACCE+SCRB
2173     0F63   CD 71 00                CALL   PUSHF
2174     0F66   2A 78 14                LHLD   PT1
2175     0F69   CD E6 0F                CALL   LOD
2176     0F6C   CD D1 0F                CALL   TST
2177     0F6F   CA A6 0F                JZ     ZFEX
2178     0F72   FC DC 03                CM     ERROR6     ;NEGATIVE
2179     0F75   1E 20                   MVI    E,32
2180     0F77   CD 8E 12                CALL   FIX
2181     0F7A   5A                      MOV    E,D
2182     0F7B   51                      MOV    D,C
2183     0F7C   EB                      XCHG
2184     0F7D   22 7A 15                SHLD   TEMP
2185     0F80   21 35 00                LXI    H,ACCE+SCRB
2186     0F83   CD 7F 00                CALL   POPF
2187     0F86   97                      SUB    A
2188     0F87   32 39 00                STA    ACC3
2189     0F8A   CD D1 0F                CALL   TST
2190     0F8D   21 6C 15                LXI    H,FLAC
2191     0F90   CD B6 0F                CALL   STR
2192     0F93   2A 7A 15        FLLP:   LHLD   TEMP
2193     0F96   2B                      DCX    H
2194     0F97   7C                      MOV    A,H
2195     0F98   B5                      ORA    L
2196     0F99   C8                      RZ
2197     0F9A   22 7A 15                SHLD   TEMP
2198     0F9D   21 6C 15                LXI    H,FLAC
2199     0FA0   CD 04 10                CALL   MUL
2200     0FA3   C3 93 0F                JMP    FLLP
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

2201     0FA6    21  6C  15      ZFEX:   LXI     H,FLAC
2202     0FA9    CD  7F  00              CALL    POPF
2203     0FAC    21  62  15              LXI     H,FLTONE
2204     0FAF    CD  E6  0F              CALL    LOD
2205     0FB2    C9                      RET
2206     0FB3    C9              LIBRARY: RET
2207
```

```
 STMNT      LOC     OBJECT CODE       SOURCE STATEMENT

 2209                                 *
 2210                                 * 8008 BINARY FLOATING POINT SYSTEM
 2211                                 *
 2212                                 * THE 8008 BINARY FLOATING POINT SYSTEM CONSISTS OF A SET OF SUBROUTINES
 2213                                 * DESIGNES TO PERFORM ARITHMETIC OPERATIONS ON NUMERIC QUANTITIES
 2214                                 * REPRESENTED IN MEMORY.
 2215                                 *
 2216                                 * EACH NUMERIC QUANTITY OCCUPIED FOUR CONSECUTIVE WORDS (32 BITS) OF
 2217                                 * MEMORY.  THE LARGEST MAGNITUDE THATN CAN BE REPRESENTED IS APPROXIMATELY
 2218                                 * 3.6 TIMES TEN TO THE 38TH POSWER.  THE SMALLEST NON-ZERO MAGNITUDE THAT
 2219                                 * CAN BE REPRESENTED IS APPROXIMATELY 2.7 TIME TEN TO THE MINUS 39TH POSWER.
 2220                                 * EACH NUMERIC QUANTITY IS REPRESENTED WITH A PRECISION OF ONE PART IN
 2221                                 * APPROXIMATELY 16,000,000.
 2222                                 *
 2223                                 * THE SOFTWARE CONSTITUTING THE FLOATING POINT SYSTEM IS DIVIDED INTO
 2224                                 * TWO SECTIONS.  EACH OF WHICH OCCUPIES
 2225                                 * 3 BANKS OF ROM OR RAM.  SECTION 1 IS INDEPENDENT OF OTHER SOFTWARE.
 2226                                 * SECTION 2 IS OPERABLE ONLY WHEN SECTION 1 IS AVAILABLE IN MEMORY.  IN
 2227                                 * ADDITION TO MEMORY REQUIRED FOR PROGRAM, 63 WORDS OF RAM ARE USED AS
 2228                                 * SCRATCHPAD.
 2229                                 *
 2230                                 * SOFTWARE SECTION 1 CONTAINS THE FOLLOWING SUBROUTINES:
 2231                                 *
 2232                                 *      LOD - LOAD SPECIFIED DATA INTO THE FLOATING POINT ACCUMULATOR.
 2233                                 *      ADD - ADD SPECIFIED DATA TO THE FLOATING POINT ACCUMULATOR.
 2234                                 *      SUB - SUBTRACT SPECIFIED DATA FROM THE FLOATING POINT ACCUMULATOR.
 2235                                 *      MUL - MULTIPLY SPECIFIED DATA TIME THE FLOATING POINT ACCUMULATOR.
 2236                                 *      DIV - DIVIDE SPECIFIED DATA INTO THE FLOATING POINT ACCUMULATOR.
 2237                                 *      TST - SET CONTROL BITS TO INDICATE ATTRIBUTES OF THE FLOATING POINT
 2238                                 *            ACCUMULATOR.
 2239                                 *      CHS - CHANGE THE SIGN OF THE FLOATING POINT ACCUMULATOR.
 2240                                 *      ABS - SET THE SIGN OF THE FLOATING POINT ACCUMULATOR POSITIVE.
 2241                                 *      STR - STORE IN SPECIFIED MEMORY THE VALUE IN THE REGISTERS AS
 2242                                 *            RETURNED BY OTHER SUBROUTINES.
 2243                                 *      INIT - MOVE CODE FROM ROM TO RAM IN PREPARATION FOR EXECUTION OF THE
 2244                                 *             MUL AND DIV SUBROUTINES.
 2245                                 *
 2246                                 * SOFTWARE SECTION 2 CONTAINS SUBROUTINES WHICH ARE USED TO CONVERT DATA
 2247                                 * BETWEEN THE BINARY FLOATING POINT FORMAT AND A DECIMAL FORMAT SUITABLE
 2248                                 * FOR ENTRY OF DISPLAY ON INPUT/OUTPUT EQUIPMENT.  THE DECIMAL FORMAT IS
 2249                                 * STORED IN MEMORY AS A SERIES OF CHARACTERS.  RELATIVELY SIMPLE INPUT/OUTPUT
 22=0                                 * ROUTINES MAY BE USED TO INTERFACE THE MEMORY RESIDENT CHARACTER STRINGS WITH
 22=1                                 * ANY TYPE OF PHYSICAL I/O DEVICE.
 22=2                                 *
 22=3                                 * THE CHARACTER STRINGS CONSIST OF BCD REPRESENTATIONS OF DECIMAL DIGITS AND
 22=4                                 * ARBITRARY REPRESENTATIONS OF +, -, . AND EXPONENTIAL SIGN (LETTER E), AND
 22=5                                 * SPACE.  CHARACTER STRINGS MAY NOT CROSS MEMORY BANK BOUNDARIES.  AN INPUT
 22=6                                 * STRING IS THEREFORE LIMITED TO 256 CHARACTERS.  AN OUTPUT STRING CONSISTS
 22=7                                 * OF 13 CHARACTERS.
 22=8                                 *
 22=9                                 * THE OUT SUBROUTINE GENERATES CHARACTER STRINGS IN 2 FORMATS: THE CHOICE
 2260                                 * OF FORMAT DEPENDS ON THE MAGNITUDE OF THE VALUE REPRESENTED.
 2261                                 *      MAGNITUDES BEWTEEN .1000000 AND 999999, ARE REPRESENTED BY A SPACE
 2262                                 *      OR MINUS SIGN, SEVEN DECIMAL DIGITS AND APPROPRIATELY POSITIONED
 2263                                 *      DECIMAL POINT, AND FOUR SPACES.
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

2264                             *      MAGNITUDES OUTSIDE THE RANGE ARE REPRESENTED BY A SPACE OR MINUS SIGN,
2265                             *      A VALUE BEWTEEN 1.000000 AND 9.999999, AN EXPONENTIAL SIGN, AND A
2266                             *      SIGNED TWO DIGIT POWER OF TEN.
2267                             *
2268                             * THE INP SUBROUTINE CONVERTS CHARACTER STRINGS IN EITHER OF THE ABOVE TWO
2269                             * FORMATS, OR A MODIFIED VERSION OF THEM.  THE LEADING SIGN MAY BE INCLUDED
2270                             * OR OMITTED.  ANY NUMBER OF DIGITS MAY BE USED TO INDICATE THE VALUE,
2271                             * WITH OR WITHOUT AN INCLUDED DECIMAL POINT.  IF A POWER-OF-TEN MULTIPLIER IS
2272                             * INDICATED IT MAY BE SIGNED OR UNSIGNED AND MAY CONTAIN ONE OR TWO DIGITS.
2273                             * AN INPUT STRING IS TERMINATED BY THE FIRST CHARACTER WHICH DEPARTS FROM THE
2274                             * THE FOLLOWING ARE EXAMPLES OF INPUT AND CORRESPONDING OUTPUT CHARACTER
2275                             * STRINGS.
2276                             *           3.141593          3.141593
2277                             *      -.0000000000001        -1.000000E-13
2278                             *           +1.6E5
2279                             *           +1.6E5            160000.0
2280                             *           123456789        11.234568E+08
2281                             *           54321E-10         5.432100E-06
2282                             *           -2718281828E9     -2.718282
2283                             *
2284                             *           8008 BINARY FLOATING POINT SYSTEM
2285                             *
2286                             *      THE 8008 BINARY FLOATING POINT SYSTEM CONSISTS OF A SET OF
2287                             * SUBROUTINES DESIGNED TO PERFORM OPERATIONS IN NUMERIC QUANTITIES
2288                             * REPRESENTED IN A SPECIFIC NOTATION.  SUBROUTINES ARE PROVIDED TO PERFORM
2289                             * A VARIETY OF ARITHMETIC AND RELATED OPERATIONS.
2290                             *
2291                             *      THE SUBROUTINES ARE DESIGNED TO BE STORED AND EXECUTED IN
2292                             * READ-ONLY-MEMORY (ROM) AND REQUIRE THE FIRST PORTION OF A BANK OF READ-WRITE-
2293                             * MEMORY(RAM) FOR SCRATCHPAD MEMORY.  THE SUBROUTINES ARE SEPARATED INTO A
2294                             * NUMBER OF PACKAGES, EACH CONTAINING SUBROUTINES FOR A GROUP OF RELATED
2295                             * OPERATIONS.  THE AMOUNT OF MEMORY (ROM AND RAM) REQUIRED FOR INSTALLATION
2296                             * OF THE SYSTEM IS DEPENDENT UPON THE COMBINATION OF PACKAGES TO BE USED.
2297                             * SCRATCHPAD MEMORY IS INITIALIZED BY A UTILITY SUBROUTINE WHICH MUST BE
2298                             * EXECUTED BEFORE OTHER SUBROUTINES ARE EXECUTED THE FIRST TIME.
2299                             *
2300                             *      IN GENERAL, THE SUBROUTINES HAVE SIMILIAR ENTRY AND EXIT CONDITIONS,
2301                             * UNLESS SPECIFIED DIFFERENTLY IN THE DESCRIPTION OF A SPECIFIC SUBROUTINE, THE
2302                             * SUBROUTINES HAVE THE FOLLOWING CHARACTERISTICS.
2303                             *
2304                             *      SUBROUTINES REQUIRING ONE OPERAND TAKE IT FROM AN INTERNAL FLOATING
2305                             * POINT ACCUMULATOR.  SUBROUTINES REQUIRING TWO OPERAND TAKE ONE FROM THE
2306                             * ACCUMULATOR AND THE OTHER FROM THE MEMORY LOCATION INDICATED BY THE CONTENTS
2307                             * OF THE H AND L REGISTERS UPON ENTRY.  THE NUMERIC RESULT OF EACH OPERATION IS
2308                             * STORED IN THE ACCUMULATOR AND IS RETURNED TO THE CALLER IN THE A, B, C, AND D
2309                             * REGISTERS.
2310                             *
2311                             * UPON EXIT FROM THE ARITHMETIC SUBROUTINES, THE PROPERTIES OF THE RESULT
2312                             * ARE INDICATED BY THE SETTINGS OF THE CONTROL BITS.
2313                             *      CARRY BIT = 1 THE RESULT EXCEEDS THE CAPACITY OF THE ACCUMULATOR.  THE
2314                             *                    OTHER CONTROL BITS, THE CONTENTS OF THE HARDWARE
2315                             *                    REGISTERS, AND THE CONTENTS OF THE ACCUMULATOR ARE
2316                             *                    MEANGINGLESS.  THIS SITUATION IS ALSO INDICATED BY A
2317                             *                    NON-ZERO QUANTITY BEING STORED IN A FLAG WORD.
2318                             *      CARRY BIT = 0 THE RESULT IS IN RANGE.  THE ZERO AND SIGN BITS ARE
```

```
STMNT     LOC     OBJECT CODE     SOURCE STATEMENT

2319                              *                    PROPERLY SET, AND THE A, B, C, AND D REGISTERS CONTAIN
2320                              *                    A REPRESENTATION OF THE VALUE OF THE ACCUMULATOR.
2321                              *       ZERO BIT = 1 THE RESULT OF THE OPERATION IS ZERO OR A QUANTITY TOO
2322                              *                    SMALL TO BE REPRESENTED.
2323                              *       ZERO BIT = 0 THE RESULT IS NON-ZERO.
2324                              *       SIGN BIT = 1 THE RESULT IS NEGATIVE
2325                              *       SIGN BIT = 0 THE RESULT IS POSITIVE
2326                              *
2327                              *     DATA ARE REPRESENTED IN A NOTATION WHICH RECORDS EIGHT BITS EXPONEN,
2328                              * ONE BIT OF SIGN, AND TWENTY FOUR BITS OF FRACTION.
2329                              * THE LARGEST MAGNITUDE THAT CAN BE REPRESENTED IS APPROXIMATELY 3.6 * 10 ** 38.
2330                              * THE SMALLES NON-ZERO MAGNITUDE IS APPROXIMATELY 2.7 * 10 ** -39.  THE
2331                              * RESOLTUION OF THE NOTATION IS APPROXIMATELY 6.2 * 10 ** -8, I.E., BETTER THAN
2332                              * SEVEN DECIMAL DIGIT PRECISION.
2333                              *
2334                              *
2335                              *     DATA VALUES ARE REPRESENTED IN FOUR CONSECTIVE MEMORY WORDS WHICH MUST
2336                              * BE IN SAME BACK OF MEMORY.  THE INTERPRETATION OF THESE WORD IS SHOWN
2337                              * BELOW.
2338                              *     WORD 1    IF NON-ZERO, THIS CONTAINS THE EXPONENT PLUS A BIAS OF
2339                              *               200 OCTAL.  THE EXPONENT INDICATES THE POWER OF 2 BY
2340                              *               WHICH THE FRACTION IS MULTIPLIED TO OBTAIN THE REPRESENTED
2341                              *               VALUE.  FI THIS WORD IS ZERO THE REPRESENTED VALUE IS ZERO
2342                              *               AND WORDS 2, 3, AND 4 ARE MEANINGLESS.
2343                              *
2344                              *     WORD2, BIT 7
2345                              *               THIS BIT INDICATES THE SIGN OF THE VALUE:
2346                              *               0 IF POSITIVE, 1 IF NEGATIVE.
2347                              *     WORD2, BITS 6-0
2348                              *               THESE BITS PLUS AN ASSUMMED 1 INT BIT 7 ARE THE MOST
2349                              *               SIGNIFICANT BITS OF THE FRACTION.  THE FRACTION IS STORED
2350                              *               IN ABSOLUTE FORM (UNSIGNED) WITH THE RADIX POINT
2351                              *               POSITIONED TO THE LEFT OF BIT 7.  THE VALUE OF THE FRACTION
2352                              *               IS THUS LESS THAN 1.0 AND EQUAL TO OR GREATER THEN 0.5.
2353                              *     WORD 3 THIS WORD CONTAINS THE SECOND MOST SIGNIFICANT EIGHT BITS OF THE
2354                              *            FRACTION.
2355                              *     WORD 4 THIS WORD CONTAINS THE LEAST SIGNIFICANT EIGHT BITS OF THE
2356                              *            FRACTION.
2357                              *
2358                              *     EXAMPLES OF DATA NOTATION.
2359                              *
2360                              *     VALUE     WORD1     WORD2     WORD3     WORD4
2361                              *      0.0       000       XXX       XXX       XXX        X = DONT CARE
2362                              *     +1.0       201       000       000       000
2363                              *     -1.0       201       200       000       000
2364                              *     +0.1       175       114       314       314
2365                              *   -100.1       207       310       063       063
2366                              *
2367                              *     FLOATING POINT ACCUMULATOR.
2368                              *
2369                              *     THE FLOATING POINT ACCUMULATOR CONSISTS OF 5 SCRATCHPAD WORDS
2370                              * CONTAINING RESPECTIVELY THE ACCUMULATOR EXPONENT, THE ACCUMULATOR SIGN,
2371                              * AND THREE WORDS OF ACCUMULATOR FRACTION.  THE EXPONENT OS RECORDED WITH A
2372                              * BIAS OF 200 OCTAL.  AN EXPONENT WORD OF ZERO INDICATES THAT THE VALUE IN THE
2373                              * ACCUMULATOR IS ZERO AND THE REMAINING WORDS OF THE ACCUMULATOR ARE MEANINGLESS
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

2374                             * THE SIGN WORD HOLDS 000 IF THE ACCUMULATOR IS NEGATIVE. 200 IF POSITIVE.  THE
2375                             * FRACTION IS RECORDED AS A NORMALIZED POSITIVE VALUE WITH THE RADIX POINT TO
2376                             * THE LEFT OF THE MOST SIGNIFICANT BIT OF THE FIRST FRACTION WORD.
2377                             *
2378                             *      OVERFLOW FLAG.
2379                             *
2380                             *      THE OVERFLOW FLAG WORD IS PROVIDED AS A CONVENIENCE TO THE USER OF THE
2381                             * FLOATING POINT SYSTEM.  THE WORD IS INITIALLY SET TO ZERO AND MAY BE RESET
2382                             * TO ZERO BY THE USER AT ANY TIME.  WHEN ANY OF THE SYSTEM ROUTINES DETECT
2383                             * AN OVERFLOW CONDITION THE OVERFLOW FLAG IS SET NON-ZERO.  THUS THE USER MAY
2384                             * CLEAR THE FLAG, PERFORM A SEQUENCE OF LOATING POINT OPERATIONS, AND CHECK THE
2385                             * FLAG TO DETERMINE IF AN OVERFLOW OCCURRED ANYWHERE IN THE SEQUENCE.
2386                             *
2387                             *      SIGNIFICANCE INDEX.
2388                             *
2389                             *      THE FLOATING POINT ADD AND SUBTRACT SUBROUTINES RETURN A SIGNIFICANCE
2390                             * INDEX TO THE USER WHEN THE RESULT OF THE OPERATION IS NOT ZERO.  THIS INDEX
2391                             * GIVES AN INDICATION OF THE CHANGE IN THE VALUE OF THE ACCUMULATOR EXPONENT AS
2392                             * A RESULT OF THE ARITHMETIC OPERATION PERFORMED.  IT IS USED PRIMARILY FOR
2393                             * COMPARISON OF TWO VALUES WHICH ARE EXPECTED TO BE EQUAL, BUT WHICH MAY DIFFER
2394                             * BY A SMALL AMOUNT DUE TO MEASUREMENT OR ROUND-OFF ERROR.  AS AN EXAMPLE, A
2395                             * SIGNIFICANCE INDEX OF 354 OCTAL (-20 DECIMAL) INDICATES THAT THE RESULT OF
2396                             * THE OPERATION IS SMALLER THAT THE OPERANDS BY A FACTOR OF APPROXIMATELY
2397                             * ONE MILLION (2 ** 20).
2398                             * THE FLOATING POINT TEST, COMPLEMENT AND ABOLSUTE SUBROUTINES RETURN THE
2399                             * SIGNIFICANCE INDEX FROM AN IMMEDIATELY PRECEEDING ADD OR SUBTRACT OPERATION.
```

```
STMNT    LOC    OBJECT CODE    SOURCE STATEMENT

2401                           *          8008 BINARY FLOATING POINT SYSTEM
2402                           *
2403                           * ARITHMETIC AND UTILITY PACKAGE
2404                           *
2405                           * THE ARITHMETIC AND UTILITY SUBROUTINE PACKAGE OF THE 8008 BINARY FLOATING
2406                           * POINT SYSTEM CONTAINS SUBROUTINES FOR PERFORMING THE BASIC ARITHMETIC AND
2407                           * UTILITY OPERATIONS AVAILABLE IN THE SYSTEM.
2408                           *
2409                           *
2410                           * STORE REGISTERS SUBROUTINE
2411                           *
2412                           * THE STORE REGISTERS SUBROUTINE STORES THE CONTENTS OF THE
2413                           * A, B, C, AND D REGISTERS IN FOUR CONSECUTIVE MEMORY LOCATIONS
2414                           * (IN THE SAME BANK OF RAM).  THE ADDRESS WHERE THE FIRST WORD
2415                           * WILL BE STORED IS INDICATED BY THE CONTENTS OF THE H AND L
2416                           * REGISTERS
2417                           *
2418                           * ENTRY POINT
2419                           *      STR
2420                           *
2421                           * ENTRY CONDITIONS
2422                           *      A REGISTER = 1ST WORD TO BE STORED
2423                           *      B REGISTER = 2ND WORD TO BE STORED
2424                           *      C REGISTER = 3RD WORD TO BE STORED
2425                           *      D REGISTER = 4TH WORD TO BE STORED
2426                           *      H REGISTER = MS 6 BITS OF MEMORY ADDRESS
2427                           *      L REGISTER = LS 8 BITS OF MEMORY ADDRESS
2428                           *
2429                           * EXIT CONDITIONS
2430                           *      THE CONTENTS OF THE REGISTERS STORED IN THE SPECIFIED
2431                           *      MEMORY LOCATION
2432                           *
2433                           * REGISTERS ALTERED
2434                           *      L
2435                           *
2436                           * MAXIMUM SUBROUTINE LEVELS USED
2437                           *      0
2438                           *
2439                           *
2440                           * FLOATING POINT LOAD SUBROUTINE
2441                           *
2442                           * THE FLOATING POINT LOAD SUBROUTINE PLACES THE SPECIFIED
2443                           * FLOATING POINT OPERAND IN THE FLOATING POINT ACCUMULATOR.
2444                           *
2445                           * ENTRY POINT
2446                           *      LOD
2447                           *
2448                           * ENTRY CONDITIONS
2449                           *      H REGISTER = 6 MS BITS OF OPERAND ADDRESS
2450                           *      L REGISTER = LS 8 BITS OF OPERAND ADDRESS
2451                           *
2452                           * EXIT CONDITIONS
2453                           *      CONTROL BITS SET AS DEFINED FOR THE SYSTEM
2454                           *      A REGISTER = ACCUMULATOR EXPONENT
2455                           *      B REGISTER = ACCUMULATOR SIGN AND 1ST FRACTION
```

```
STMNT   LOC   OBJECT CODE   SOURCE STATEMENT

2456                        *       C REGISTER = ACCUMULATOR 2ND FRACTION
2457                        *       D REGISTER = ACCUMULATOR 3RD FRACTION
2458                        *
2459                        * REGISTERS ALTERED
2460                        *       ALL
2461                        *
2462                        * MAXIMUM SUBROUTINE LEVELS USED
2463                        *       1
2464                        *
2465                        *
2466                        * FLOATING POINT ADD SUBROUTINE
2467                        *
2468                        * THE FLOATING POINT ADD SUBROUTINE ADDS THE SPECIFIED
2469                        * FLOATING POINT OPERAND TO THE VALUE IN THE FLOATING POINT
2470                        * ACCUMULATOR AND PLACES THE SUM IN THE FLOATING POINT
2471                        * ACCUMULATOR.
2472                        *
2473                        * ENTRY POINT
2474                        *       ADD
2475                        *
2476                        * ENTRY CONDITIONS
2477                        *       H REGISTER = MS 6 BITS OF OPERAND ADDRESS
2478                        *       L REGISTER = LS 8 BITS OF OPERAND ADDRESS
2479                        *
2480                        * EXIT CONDITIONS
2481                        *       IF OVERFLOW
2482                        *          LOCATION -OVER- SET NON-ZERO
2483                        *       IF NO OVERFLOW
2484                        *       CONTROL BITS SET AS DEFINED FOR THE SYSTEM
2485                        *          A REGISTER = ACCUMULATOR EXPONENT
2486                        *          B REGISTER = ACCUMULATOR SIGN AND 1ST FRACTION
2487                        *          C REGISTER = ACCUMULATOR 2ND FRACTION
2488                        *          D REGISTER = ACCUMULATOR 3RD FRACTION
2489                        *          E REGISTER = SIGNIFICANCE INDEX
2490                        *
2491                        * REGISTERS ALTERED
2492                        *       ALL
2493                        *
2494                        * MAXIMUM SUBROUTINES LEVLES USED
2495                        *       2
2496                        *
2497                        *
2498                        * FLOATING POINT SUBTRACT SUBROUTINE
2499                        *
2500                        * THE FLOATING POINT SUBTRACT SUBROUTINE SUBTRACTS THE SPECIFIED
2501                        * FLOATING POINT OPERAND FROM THE VALUE IN THE FLOATING POINT
2502                        * ACCUMULATOR AND PLACES THE DIFFERENCE IN THE FLOATING POINT
2503                        * ACCUMULATOR.
2504                        *
2505                        * ENTRY POINT
2506                        *       SUB
2507                        *
2508                        * ENTRY CONDITIONS
2509                        *       H REGISTER = MS 6 BITS OF OPERAND ADDRESS
2510                        *       L REGISTER = LS 8 BITS OF OPERAND ADDRESS
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

2511                             *
2512                             * EXIT CONDITIONS
2513                             *      CONTROL BITS SET AS DEFINED FOR THE SYSTEM
2514                             *      IF OVERFLOW
2515                             *         LOCATION -OVER- SET NON-ZERO
2516                             *      IF NO OVERFLOW
2517                             *         A REGISTER = ACCUMULATOR EXPONENT
2518                             *         B REGISTER = ACCUMULATOR SIGN AND 1ST FRACTION
2519                             *         C REGISTER = ACCUMULATOR 2ND FRACTION
2520                             *         D REGISTER = ACCUMULATOR 3RD FRACTION
2521                             *         E REGISTER = SIGNIFICANCE INDEX
2522                             *
2523                             * REGISTERS ALTERED
2524                             *      ALL
2525                             *
2526                             * MAXIMUM SUBROUTINE LEVLES USED
2527                             *      2
2528                             *
2529                             *
2530                             * FLOATING POINT MULTIPLY SUBROUTINE
2531                             *
2532                             * THE FLOATING POINT MULTIPLY SUBROUTINE MULTIPLIES THE
2533                             * SPECIFIED FLOATING POINT OPERAND BY THE VALUE IN THE FLOATING
2534                             * POINT ACCUMULATOR AND PLACES THE PRODUCT IN THE FLOATING
2535                             * POINT ACCUMULATOR.
2536                             *
2537                             * ENTRY POINT
2538                             *      MUL
2539                             *
2540                             * ENTRY CONDITIONS
2541                             *      H REGISTER = MS 6 BITS OF OPERAND ADDRESS
2542                             *      L REGISTER = LS 8 BITS OF OPERAND ADDRESS
2543                             *
2544                             * EXIT CONDITIONS
2545                             *      CONTROL BITS SET AS DEFINED FOR THE SYSTEM
2546                             *      IF OVERFLOW
2547                             *         LOCATION -OVER- SET NONZERO
2548                             *      IF NO OVERFLOW
2549                             *         A REGISTER = ACCUMULATOR EXPONENT
2550                             *         B REGISTER = ACCUMULATOR SIGN AND 1ST FRACTION
2551                             *         C REGISTER = ACCUMULATOR 2ND FRACTION
2552                             *         D REGISTER = ACCUMULATOR 3RD FRACTION
2553                             *
2554                             * REGISTERS ALTERED
2555                             *      ALL
2556                             *
2557                             * MAXIMUM SUBROUTINE LEVELS USED
2558                             *      2
2559                             *
2560                             *
2561                             * FLOATING POINT DIVIDE SUBROUTINE
2562                             *
2563                             * THE FLOATING POINT DIVIDE SUBROUTINE DIVIDES THE SPECIFIED
2564                             * FLOATING POINT OPERAND INTO THE VALUE IN THE FLOATING POINT
2565                             * ACCUMULATOR AND PLACES THE QUOTIENT IN THE FLOATING POINT
```

```
STMNT   LOC   OBJECT CODE     SOURCE STATEMENT

2566                          * ACCUMULATOR.
2567                          *
2568                          * ENTRY POINT
2569                          *      DIV
2570                          *
2571                          * ENTRY CONDITIONS
2572                          *      H REGISTER = MS 6 BITS OF OPERAND ADDRESS
2573                          *      L REGISTER = LS 8 BITS OF OPERAND ADDRESS
2574                          *
2575                          * EXIT CONDITIONS
2576                          *      CONTROL BITS SET AS DEFINED FOR THE SYSTEM
2577                          *      IF OVERFLOW
2578                          *         LOCATION -OVER- SET NON-ZERO
2579                          *      IF NO OVERFLOW
2580                          *         A REGISTER = ACCUMULATOR EXPONENT
2581                          *         B REGISTER = ACCUMULATOR SIGN AND 1ST FRACTION
2582                          *         C REGISTER = ACCUMULATOR 2ND FRACTION
2583                          *         D REGISTER = ACCUMULATOR 3RD FRACTION
2584                          *
2585                          * REGISTERS ALTERED
2586                          *      ALL
2587                          *
2588                          * MAXIMUM SUBROUTINE LEVELS USED
2589                          *      2
2590                          *
2591                          *
2592                          * FLOATING POINT ABSOLUTE SUBROUTINE
2593                          *
2594                          * THE FLOATING POINT ABSOLUTE SUBROUTINE SETS THE SIGN OF THE
2595                          * VALUE IN THE FLOATING POINT ACCUMULATOR POSITIVE.
2596                          *
2597                          * ENTRY POINT
2598                          *      ABS
2599                          *
2600                          * ENTRY CONDITIONS
2601                          *      NONE
2602                          *
2603                          * EXIT CONDITIONS
2604                          *      CONTROL BITS SET AS DEFINED FOR THE SYSTEM
2605                          *         A REGISTER = ACCUMULATOR EXPONENT
2606                          *         B REGISTER = ACCUMULATOR SIGN AND 1ST FRACTION
2607                          *         C REGISTER = ACCUMULATOR 2ND FRACTION
2608                          *         D REGISTER = ACCUMULATOR 3RD FRACTION
2609                          *      E REGISTER = SIGNIFICANCE INDEX. IF THE PREVIOUS
2610                          *                   OPERATION WAS AN ADD OR SUBTRACT
2611                          *
2612                          * REGISTERS ALTERED
2613                          *      ALL
2614                          *
2615                          * MAXIMUM SUBROUTINE LEVELS USED
2616                          *      0
2617                          *
2618                          *
2619                          * FLOATING POINT ZERO SUBROUTINE
2620                          *
```

```
STMNT    LOC    OBJECT CODE    SOURCE STATEMENT

2621                           * THE FLOATING POINT ZERO SUBROUTINE PLACES THE VALUE ZERO IN
2622                           * THE FLOATING POINT ACCUMULATOR.
2623                           *
2624                           * ENTRY POINT
2625                           *      ZRO
2626                           *
2627                           * ENTRY CONDITIONS
2628                           *      NONE
2629                           *
2630                           * EXIT CONDITIONS
2631                           *      CONTROL BITS SET AS DEFINED FOR THE SYSTEM
2632                           *         A REGISTER = ACCUMULATOR EXPONENT
2633                           *         B REGISTER = ACCUMULATOR SIGN AND 1ST FRACTION
2634                           *         C REGISTER = ACCUMULATOR 2ND FRACTION
2635                           *         D REGISTER = ACCUMULATOR 3RD FRACTION
2636                           *
2637                           * REGISTERS ALTERED
2638                           *      A,H,L
2639                           *
2640                           * MAXIMUM SUBROUTINE LEVELS USED
2641                           *      0
2642                           *
2643                           *
2644                           * FLOATING POINT TEST SUBROUTINE
2645                           *
2646                           * THE FLOATING POINT TEST SUBROUTINE LOADS THE VALUE IN THE
2647                           * FLOATING POINT ACCUMULATOR INTO THE REGISTERS AND SETS THE
2648                           * ZERO AND SIGN CONTROL BITS TO INDICATE THE CORRESPONDING
2649                           * ATTRIBUTES OF THE VALUE.
2650                           * ENTRY POINT
2651                           *      TST
2652                           *
2653                           * ENTRY CONDITIONS
2654                           *      NONE
2655                           *
2656                           * EXIT CONDITIONS
2657                           *      CONTROL BITS SET AS DEFINED FOR THE SYSTEM
2658                           *         A REGISTER = ACCUMULATOR EXPONENT
2659                           *         B REGISTER = ACCUMULATOR SIGN AND 1ST FRACTION
2660                           *         C REGISTER = ACCUMULATOR 2ND FRACTION
2661                           *         D REGISTER = ACCUMULATOR 3RD FRACTION
2662                           *      E REGISTER = SIGNIFICANCE INDEX, IF THE PREVIOUS
2663                           *                   OPERATION WAS AN ADD OR SUBTRACT
2664                           *
2665                           * REGISTERS ALTERED
2666                           *      ALL
2667                           *
2668                           * MAXIMUM SUBROUTINE LEVELS USED
2669                           *      0
2670                           *
2671                           *
2672                           * FLOATING POINT COMPLEMENT SUBROUTINE
2673                           *
2674                           * THE FLOATING POINT COMPLEMENT SUBROUTINE CHANGES THE
2675                           * ARITHMETIC SIGN OF THE VALUE IN THE FLOATING POINT
```

```
STMNT     LOC     OBJECT CODE       SOURCE STATEMENT

2676                                * ACCUMULATOR.
2677                                * ENTRY POINT
2678                                *      CHS
2679                                *
2680                                * ENTRY CONDITIONS
2681                                *      NONE
2682                                *
2683                                * EXIT CONDITIONS
2684                                *      CONTROL BITS SET AS DEFINED FOR THE SYSTEM
2685                                *         A REGISTER = ACCUMULATOR EXPONENT
2686                                *         B REGISTER = ACCUMULATOR SIGN AND 1ST FRACTION
2687                                *         C REGISTER = ACCUMULATOR 2ND FRACTION
2688                                *         D REGISTER = ACCUMULATOR 3RD FRACTION
2689                                *      E REGISTER = SIGNIFICANCE INDEX. IF THE PREVIOUS
2690                                *                   OPERATION WAS AN ADD OR SUBTRACT
2691                                *
2692                                * REGISTERS ALTERED
2693                                *      ALL
2694                                *
2695                                * MAXIMUM SUBROUTINE LEVELS USED
2696                                *      0
2697                                *
```

```
STMNT      LOC     OBJECT CODE       SOURCE STATEMENT

2699      0FB4                      ARITH:    EQU     $
2700      000F                      ARTHB:    EQU     <ARITH
2701                                ;     8008 BINARY FLOATING POINT SYSTEM
2702                                ;
2703                                ;     MODIFIED FOR 8080 FALL 1975
2704                                ;
2705                                ;     ARITHMETIC AND UTILITY PACKAGE
2706                                ;     PROGRAMMMER: CAL OHME
2707                                ;     SUPPLIED FROM INTEL'S USER LIBRARY
2708                                ;     DATE  26 DECEMBER 1973
2709                                ;     ARITH IS THE BEGGINING ADDRESS OF THE
2710                                ;     ARITHMETIC AND UTILITY PACKAGE OF THE FLOATING
2711                                ;     POINT SYSTEM.
2712                                ;     SCR IS THE BEGINNING ADDRESS OF THE
2713                                ;     RAM USED AS SCRATCHPAD FOR THE SYSTEM
2714                                ;     THE RAM MULTIPLY AND DIVIDE SUBROUTINES
2715                                ;     ARE MOVED FROM ROM TO RAM BY SUBROUTINE
2716                                ;     INIT AND ARE EXECUTED IN RAM ONLY.
2717                                ;     RAM MULTIPLY SUBROUTINE.
2718                                ;     INIT SUBROUTINE ENTRY POINT
2719      0FB4                      INIT:     EQU     $
2720                                ;
2721                                ;
2722                                ;     STR SUBROUTINE ENTRY POINT
2723      0FB4      73              STR0:     MOV     M,E       ;STORE ZEROETH WORD
2724      0FB5      2C              INR       L         ;TO ADDRESS FIRST WORD
2725      0FB6      77              STR:      MOV     M,A       ;STORE FIRST WORD
2726      0FB7      2C 23   INXH    STR1:     INR     L         ;TO ADDRESS SECOND WORD
2727      0FB8      70                        MOV     M,B       ;STORE SECOND WORD
2728      0FB9      2C 23   INXH              INR     L         ;TO ADDRESS THIRD WORD
2729      0FBA      71                        MOV     M,C       ;TO STORE THIRD WORD
2730      0FBB      2C 23   INXH              INR     L         ;TO ADDRESS FOURTH WORD
2731      0FBC      72                        MOV     M,D       ;STORE FOURTH WORD
2732      0FBD      C9                        RET               ;RETURN TO CALLER
2733                                ;     FLOATING POINT ZERO SUBROUTINE ENT. PNT.
2734      0FBE      26  00          ZR0:      MVI     H,SCRB    ;TO ADDRESS SCRATCH BANK
2735      0FC0      2E  35          ZR01:     MVI     L,ACCE    ;TO ADDR ACCUM EXPONENT
2736      0FC2      AF                        XRA     A         ;ZERO
2737      0FC3      77                        MOV     M,A       ;CLEAR ACCUMULATOR EXPONENT
2738      0FC4      C9                        RET               ;RETURN TO CALLER
2739                                ;     FLOATING POINT CHS SUBROUTINE ENT. PNT.
2740      0FC5      3E  80          CHS:      MVI     A,2000    ;MASK FOR SIGN BIT
2741      0FC7            06                  DB      0060      ;LBI INST TO SKIP NEXT WD
2742                                ;     FLOATING POINT ABS SUBROUTINE ENT. PNT.
2743      0FC8      AF              ABS:      XRA     A         ;ZERO
2744      0FC9      26  00                    MVI     H,SCRB    ;TO ADDRESS SCRATCH BANK
2745      0FCB      2E  36                    MVI     L,ACCS    ;TO ADDRESS ACCUM SIGN
2746      0FCD      A6                        ANA     M         ;COMPLEMENT OF SIGN
2747      0FCE      EE  80                    XRI     2000      ;COMPLEMENT THE SIGN BIT
2748      0FD0      77                        MOV     M,A       ;ACCUMALOTOR SIGN
2749                                ;     FLOATING POINT TEST ENTRY POINT.
2750      0FD1      26  00          TST:      MVI     H,SCRB    ;TO ADDRESS SCRATCH BANK
2751      0FD3      2E  35          TST1:     MVI     L,ACCE    ;TO ADDR ACCUM EXPONENT
2752      0FD5      7E                        MOV     A,M       ;ACCUMULATOR EXPONENT
2753      0FD6      A7                        ANA     A         ;SET CONTROL BITS
```

```
STMNT   LOC   OBJECT CODE     SOURCE STATEMENT
2754    0FD7  CA BE 0F              JZ      ZRO         ;IF ACCUMALOTOR IS ZERO
2755    0FDA  5F                    MOV     E,A         ;ACCUM EXPONENT
2756    0FDB  2C                    INR     L           ;TO ADDR ACCUM SIGN
2757    0FDC  7E                    MOV     A,M         ;ACCUM SIGN
2758    0FDD  2C                    INR     L           ;TO ADDR ACCUM 1ST FRACTION
2759    0FDE  AE                    XRA     M           ;ACCUM SIGN AND 1ST FRACTION
2760    0FDF  2C                    INR     L           ;TO ADDR ACCUM 2ND FRACTION
2761    0FE0  4E                    MOV     C,M         ;ACCUMULATOR 2ND FRACTION
2762    0FE1  2C                    INR     L           ;TO ADDR ACCUM 3RD FRACTION
2763    0FE2  56                    MOV     D,M         ACCUM 3RD FRACTION
2764    0FE3  C3 F2 10              JMP     ADD12       ;TO SET EXIT CONDITIONS
2765                        ;       FLOATING POINT LOAD ENTRY POINT
2766    0FE6  7E            LOD:    MOV     A,M         ;OPERAND EXPONENT
2767    0FE7  A7                    ANA     A           ;SET CONTROL FLAGS
2768    0FE8  CA BE 0F              JZ      ZRO         ;IF OPERAND IS ZERO
2769    0FEB  5F                    MOV     E,A         ;OPERAND EXPONENT
2770    0FEC  2C-23         INXH    INR     L           ;TO ADDR OP SIGN AND 1ST
2771    0FED  7E                    MOV     A,M         ;OPERAND SIGN AND 1ST DIGIT
2772    0FEE  2C-23         INXH    INR     L           ;TO ADDRESS OPERAND 2ND DIGIT
2773    0FEF  4E                    MOV     C,M         ;OPERAND 2ND FRACTION
2774    0FF0  2C 23         INXH    INR     L           ;TO ADDRESS OPERAND 3RD FRACTION
2775    0FF1  56                    MOV     D,M         ;OPERAND 3RD FRACTION
2776                        ;       STORE THE OPERAND IN THE ACCUMULATOR
2777    0FF2  6F                    MOV     L,A         ;OPERAND SIGN AND 1ST FRACTION
2778    0FF3  F6 80         LOD1:   ORI     2000        ;ACCUMULATOR 1ST FRACTION
2779    0FF5  47                    MOV     B,A         ;ACCUM 1ST FRACTIN
2780    0FF6  AD                    XRA     L           ;ACCUM SIGN
2781    0FF7  26 00                 MVI     H,SCRB      ;TO ADDRESS SCRATCH BANK
2782    0FF9  2E 35                 MVI     L,ACCE      ;TO ADDR ACCUM EXPONENT
2783    0FFB  CD B4 0F              CALL    STRO        ;SET THE ACCUM
2784    0FFE  A8                    XRA     B           ;ACCUM SIGN AND 1ST FRACTION
2785                        ;       SET CONTROL BITS AND EXIT
2786    0FFF  47                    MOV     B,A         ;ACCUM SIGN AND 1ST FRACTION
2787    1000  F6 01                 ORI     1           ;SET SIGN BIT FOR EXIT
2788    1002  7B                    MOV     A,E         ;ACCUM EXPONENT
2789    1003  C9                    .RET                ;RETURN TO CALLER
2790                        ;       FLOATING POINT MUL SUBROUTINE ENTRY POINT.
2791    1004  7E            MUL:    MOV     A,M         ;OPERAND EXPONENT
2792    1005  A7                    ANA     A           ;SET CONTROL FLAG
2793    1006  C4 0D 11              CNZ     MDEX        ;READ OPERAND IF NOT ZERO
2794    1009  CA BE 0F              JZ      ZRO         ;IF ZERO OR UNDERFLOW
2795    100C  DA 42 10              JC      OVERF       ;IF OVERFLOW
2796    100F  CD C5 11              CALL    MULX        ;CALL FIXED MULT SUBRTN
2797                        ;       NORMALIZE IF NECESSARY
2798    1012  78                    MOV     A,B         ;1ST PRODUCT
2799    1013  A7                    ANA     A           ;SET CONROL BITS
2800    1014  FA 21 10              JM      RNDA        ;IF NO NORMALIZATION REQUIRED
2801    1017  2E 35                 MVI     L,ACCE      ;TO ADDR ACCUM EXPONENT
2802    1019  7E                    MOV     A,M         ;ACCUM EXPONENT
2803    101A  DE 01                 SBI     1           ;DECREMENT ACCUM EXPONENT
2804    101C  77                    MOV     M,A         ;ACCUM EXPONENT
2805    101D  C8                    RZ                  ;RETURN TO CALLER IF UNDERFLOW
2806    101E  CD 34 11              CALL    LSH         ;CALL LEFT SHIFT SUBROUTINE
2807                        ;       ROUND IF NECESSARY.
2808    1021  CD A8 11      RNDA:   CALL    ROND        ;CALL ROUNDING SUBROUTINE
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

2809    1024  DA 42 10              JC      OVERF     ;IF OVERFLOW
2810    1027  47                    MOV     B,A       ;ACCUM SIGN AND 1ST FRACTION
2811    1028  F6 01                 ORI     1         ;SET SIGN BIT
2812    102A  7B                    MOV     A,E       ;ACCUM EXPOENT
2813    102B  C9                    RET               ;RETURN TO CALLER
2814                        ;     FLOATING POINT DIV SUBROUTINE ENTRY POINT.
2815    102C  AF            DIV:    XRA     A         ;ZERO
2816    102D  96                    SUB     M         ;COMPLEMENT OF DIVISOR EXPONENT
2817    102E  FE 01                 CPI     1         ;SET CARRY IF DIVISION BY ZERO
2818    1030  D4 0D 11              CNC     MDEX      ;READ OPERNAD IF NOT ZERO
2819    1033  DA 42 10              JC      OVERF     ;IF OVERFLOW OR DIVISOON BY ZERO
2820    1036  CA C0 0F              JZ      ZR01      ;IF UNDERFLOW
2821    1039  4F                    MOV     C,A       ;DIVISOR 1ST FRACTION
2822    103A  CD 08 12              CALL    DIVX      ;CALL FIXED DIV SUBRTN
2823    103D  26 C0                 MVI     H,SCRB    ;TO ADDR SCRATCH BANK
2824    103F  DA 21 10              JC      RNDA      ;IF NO UNDERFLOW
2825                        ;     SET OVERFLOW FLAG
2826    1042  26 00         OVERF: MVI     H,SCRB    ;TO ADDR SCRATCH BANK
2827    1044  2E 33                 MVI     L,OVER    ;TO ADDR OVERFLOW FLAG
2828    1046  3E FF                 MVI     A,3770    ;OVERFLOW FLAG
2829    1048  77                    MOV     M,A       ;OVERFLOW FLAG
2830    1049  07                    RLC               ;SET CARRY BIT FOR EXIT
2831    104A  C9                    RET               ;RETURN TO CALLER
2832    104B              00        DB      0         ;CHECK SUM WORD
2833                        ;     FLOATING POINT SUB SUBROUTINE ENTRY POINT
2834    104C  3E 80         SB:     MVI     A,2000    ;MASK TO CHANGE OP SIGN
2835    104E              06        DB      0060      ;LBI INST TO SKIP NEXT WD
2836                        ;     FLOATING POINT ADD SUBROUTINE ENTRY POINT
2837    104F  AF            AD:     XRA     A         ;ZERO
2838                        ;     LOAD THE OPERAND
2839    1050  5E                    MOV     E,M       ;OPERAND EXPONENT
2840    1051  2C               INXH INR     L         ;TO ADDR OP SIGN 1ST FRACTION
2841    1052  AE                    XRA     M         ;OPERAND SIGN AND 1ST FRACTION
2842    1053  47                    MOV     B,A       ;OPERAND SIGN AND 1ST FRACTION
2843    1054  2C               INXH INR     L         ;TO ADDR OPERAND 2ND
2844    1055  4E                    MOV     C,M       ;OPERAND 2ND DIGIT
2845    1056  2C               INXH INR     L         ;TO ADDR OPERAND 3RD FRACTION
2846    1057  56                    MOV     D,M       ;OPERAND 3RD FRACTION
2847                        ;     SAVE INITIAL EXPONENT
2848    1058  26 00                 MVI     H,SCRB    ;TO ADDR SCRATCH PAD
2849    105A  2E 35                 MVI     L,ACCE    ;TO ADDR ACCUM EXPONENT
2850    105C  7E                    MOV     A,M       ;ACCUMALOTR EXPONENT
2851    105D  2D               DCXH DCR     L         ;TO ADDR INITIAL EXPONENT
2852    105E  77                    MOV     M,A       ;INITIAL EXPONENT
2853                        ;     CHECK FOR ZERO OPERAND
2854    105F  7B                    MOV     A,E       ;OPERAND EXPONENT
2855    1060  A7                    ANA     A         ;SET CONTROL BITS
2856    1061  CA D3 0F              JZ      TST1      ;IF OPERAND IS ZERO
2857                        ;     GENERATE SUBTRACTION FLAG, RESTORE
2858                        ;     SUPPRESSED FRACTION BIT
2859    1064  68                    MOV     L,B       ;OPERAND SIGN AND FIRST FRACTION
2860    1065  78                    MOV     A,B       ;OPERAND SIGN AND FIRST FRACTION
2861    1066  F6 80                 ORI     2000      ;OPERAND 1ST FRACTION
2862    1068  47                    MOV     B,A       ;OPERAND 1ST FRACTION
2863    1069  AD                    XRA     L         ;OPERAND SIGN
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

2864    106A    2E 36                    MVI     L,ACCS      ;TO ADDR ACCUM SIGN
2865    106C    AE                       XRA     M           ;SUBTRACTION FLAG
2866    106D    2E 3A                    MVI     L,SF        ;TO ADDR SUBRRACTION FLAG
2867    106F    77                       MOV     M,A         ;SUBTRACTION FLAG
2868                            ;    DETERMINE RELATIVE MAGNITUDES OF
2869                            ;    OPERAND AND ACCUM
2870    1070    2E 35                    MVI     L,ACCE      ;TO ADDR ACCUM EXPONENT
2871    1072    7E                       MOV     A,M         ;ACCUM EXPONENT
2872    1073    A7                       ANA     A           ;SET CONTROL BITS
2873    1074    CA FE 10                 JZ      ADD17       ;IF ACCUM IS ZERO
2874    1077    93                       SUB     E           ;DIFFERENCE IN EXPONENTS
2875    1078    DA 86 10                 JC      ADD2        ;IF ACCUM SMALLER THAN OP
2876                            ;    CHECK FOR INSIGNIFICANT OPERAND
2877    107B    FA D3 0F                 JM      TST1        ;IF OPERAND IS INSIGNIFICANT
2878    107E    FE 19                    CPI     0310        ;COMPARE SHIFT COUNT TO 25
2879    1080    DA A5 10                 JC      ADD3        ;JOIN EACH PATH IF OP SIGNIF
2880    1083    C3 D3 0F                 JMP     TST1        ;OPERAND IS INSIGNIFICANT
2881                            ;    CHECK FOR INSIGNIFICANT ACCUMULATOR
2882    1086    F2 FE 10        ADD2:    JP      ADD17       ;IF ACCUM IS INSIGNIFICANT
2883    1089    FE E7                    CPI     3470        COMPARE SHIFT COUNT TO MINUS 25
2884    108B    DA FE 10                 JC      ADD17       ;IF ACCUM IS INSIGNIFICANT
2885    108E    73                       MOV     M,E         ;OPERAND EXPONENT
2886    108F    5F                       MOV     E,A         ;SHIFT COUNT
2887    1090    2E 3A                    MVI     L,SF        ;TO ADDRESS THE SUBTRACTION FLAG
2888    1092    7E                       MOV     A,M         ;SUBTRACTION FLAG
2889    1093    2E 36                    MVI     L,ACCS      ;TO ADDR THE ACCUM SIGN
2890    1095    AE                       XRA     M           ;OPERAND SIGN
2891    1096    77                       MOV     M,A         ;ACCUM SIGN
2892    1097    AF                       XRA     A           ;ZERO
2893    1098    93                       SUB     E           ;COMPLEMENT SHIFT OUNT
2894                            ;    EXCHANGE ACCUMULATOR AND OPERAND
2895    1099    2C                       INR     L           ;TO ADDR ACCUM 1ST FRACTION
2896    109A    5E                       MOV     E,M         ;ACCUM 1ST FRACTION
2897    109B    70                       MOV     M,B         ;OPERAND 1ST FRACTION
2898    109C    43                       MOV     B,E         ;ACCUM 1ST FRACTION
2899    109D    2C                       INR     L           ;TO ADDR 2ND FRACTION
2900    109E    5E                       MOV     E,M         ;ACCUMULATOR 2ND FRACTION
2901    109F    71                       MOV     M,C         ;OPERAND 2ND FRACTION
2902    10A0    4B                       MOV     C,E         ;ACCUM 2ND FRACTION
2903    10A1    2C                       INR     L           ;TO ADDR ACCUM 3RD FRACTION
2904    10A2    5E                       MOV     E,M         ;ACCUM 3RD FRACTION
2905    10A3    72                       MOV     M,D         ;OPERAND 3RD FRACTION
2906    10A4    53                       MOV     D,E         ;ACCUM 3RD FRACTION
2907                            ;    POSITION THE OPERAND
2908    10A5    CD 41 11        ADD3:    CALL    RSH         ;POSITION THE OPERAND
2909    10A8    2E 3A                    MVI     L,SF        ;TO ADDR SUBTRACTION FLAG
2910    10AA    7E                       MOV     A,M         ;SUBTRACTION FLAG
2911    10AB    A7                       ANA     A           ;SET CONTROL BITS
2912    10AC    2E 39                    MVI     L,ACC3      ;TO ADDR ACCUM 3RD FRACTION
2913    10AE    FA D5 10                 JM      ADD9        ;IF SUBTRACTION REQUITRED
2914                            ;    ADD ADDEND TO AUGEND
2915    10B1    7E                       MOV     A,M         ;AUGEND 3RD FRACTION
2916    10B2    82                       ADD     D           ;AUGNED 3RD FRACTIN
2917    10B3    57                       MOV     D,A         ;SUM 3RD FRACTION
2918    10B4    2D                       DCR     L           ;TO ADDR AUGEND 2ND FRACTION
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

2919     10B5   7E               MOV     A,M       ;AUGNED 2ND FRACTION
2920     10B6   89               ADC     C         ;ADDEND 2ND FRACTION
2921     10B7   4F               MOV     C,A       ;SUM 2ND FRACTION
2922     10B8   2D               DCR     L         ;TO ADDR AUGNND 1ST FRACTION
2923     10B9   7E               MOV     A,M       ;AUGEND 1ST FRACTION
2924     10BA   88               ADC     B         ;ADDEND 1ST FRACTION
2925     10BB   47               MOV     B,A       ;SUM 1ST FRACTION
2926     10BC   D2  EC  10       JNC     ADD11     ;IF NO CARRY FROM 1ST DIGIT
2927                      ;      RIGHT SHIFT SUM TO NORMALIZED POISITION
2928     10BF   1F               RAR               ;RIGHT SHIFT SUM 1ST DIGIT
2929     10C0   47               MOV     B,A       ;SUM 1ST FRACTION
2930     10C1   79               MOV     A,C       ;SUM 2ND FRACTION
2931     10C2   1F               RAR               ;RIGHT SHIFT SUM 2ND DIGIT
2932     10C3   4F               MOV     C,A       ;SUM 2ND FRACTIOM
2933     10C4   7A               MOV     A,D       ;SUM 3RD FRATION
2934     10C5   1F               RAR               ;;RIGHT SHIFT SUM 3RD FRACTION
2935     10C6   57               MOV     D,A       ;SUM 3RD FRACTION
2936     10C7   1F               RAR               ;4TH FRACTION=LOW BIT OF 3RD
2937     10C8   5F               MOV     E,A       ;SUM 4TH FRACTION
2938     10C9   2E  35           MVI     L,ACCE    ;TO ADDRESS ACCUM EXPONENT
2939     10CB   7E               MOV     A,M       ;ACCUM EXPONENT
2940     10CC   C6  01           ADI     1         ;INCREMENT ACCUM EXPONENT
2941     10CE   DA  42  10       JC      OVERF     ;IF OVERVI OW
2942     10D1   77               MOV     M,A       ;ACCUM EXPONENT
2943     10D2   C3  EC  10       JMP     ADD11     ;TO ROUND FRACTION
2944                      ;      SUBTRACT SUBTRAHEND VROM MINUEND
2945     10D5   AF       ADD9:   XRA     A         ;MINUEND 4TH FRACTION IS ZERO
2946     10D6   93               SUB     E         ;SUBTRAHEND 4TH FRACTION
2947     10D7   5F               MOV     E,A       ;DIFFERENCE 4TH FRACTION
2948     10D8   7E               MOV     A,M       ;MINUEND 3RD FRACTION
2949     10D9   9A               SBB     D         ;SUBTRAHEND 3RD FRACTION
2950     10DA   57               MOV     D,A       ;DIFFERENCE 3RD FRACTION
2951     10DB   2D               DCR     L         ;TO ADDRESS MINUEND 2ND FRACTION
2952     10DC   7E               MOV     A,M       ;MINUEND 2ND FRACTION
2953     10DD   99               SBB     C         ;SUBTRAHEND 2ND FRACTION
2954     10DE   4F               MOV     C,A       ;DIFFERENCE 2ND FRACTION
2955     10DF   2D               DCR     L         ;TO ADDRESS MINUEND 1ST FRACTION
2956     10E0   7E               MOV     A,M       ;MINUEND 1ST FRACTION
2957     10E1   98               SBB     B         ;SUBTRAHEND 1ST FRACTION
2958     10E2   47               MOV     B,A       ;DIFFERENCE 1ST FRACTION
2959     10E3   DC  67  11       CC      COMP      ;COMPLEMENT IF NEGATIVE
2960     10E6   F4  7A  11       CP      NORM      ;NORMALIZE IF NECESSARY
2961     10E9   F2  C0  0F       JP      ZRO1      ;IF UNDERFLOW OR ZERO
2962     10EC   CD  A8  11       CALL    ROND      ;CALL ROUNDING ROUTINE
2963     10EF   DA  42  10       JC      OVERF     ;IF OVERFLOW
2964     10F2   47       ADD12:  MOV     B,A       ;ACCUM SIGN AND 1ST FRACTION
2965     10F3   2E  34           MVI     L,PREX    ;TO ADDRESS PREV EXPONENT
2966     10F5   7B               MOV     A,E       ;ACCUM EXPONENT
2967     10F6   96               SUB     M         ;DIFFERENCE IN EXPONENTS
2968     10F7   6F               MOV     L,A       ;DIFFERENCE IN EXPONENTS
2969     10F8   78               MOV     A,B       ;ACCUM SIGN AND 1ST FRACTION
2970     10F9   F6  01           ORI     1         ;SET SIGN BIT FOR EXIT
2971     10FB   7B               MOV     A,E       ;ACCUMULATOR EXPONENT
2972     10FC   5D               MOV     E,L       ;SIGNIFICANCE INDEX
2973     10FD   C9               RET               ;RETURN TO CALLER
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

2974                          ;     LOAD THE ACCUMULATOR WITH THE OPERAND
2975    10FE  2E 3A           ADD17:  MVI   L,SF      ;TO ADDR SUBTRACTION FLAG
2976    1100  7E                      MOV   A,M       ;SUBTRACTION FLAG
2977    1101  2E 36                   MVI   L,ACCS    ;TO ADDR ACCUMULATOR SIGN
2978    1103  AE                      XRA   M         ;OPERAND SIGN
2979    1104  2D                      DCR   L         ;TO ADDR ACCUM EXPONENT
2980    1105  CD B4 0F                CALL  STR0      ;SET THE ACUM
2981    1108  A8                      XRA   B         ;ACCUM SIGN AND FIRST FRACTION
2982    1109  C3 F2 10                JMP   ADD12     ;JOIN EXIT CODE
2983    110C        00                DB    0         ;CHECKSUM WORD
2984                          ;     SUBROUTINE TO READ THE OPERAND AND
2985                          ;     CHECK THE ACCUMULATOR EXPONENT
2986    110D  47              MDEX:   MOV   B,A       ;EXPONENT MIDIFIER
2987    110E  2C                INX   INR   L         ;TO ADDR OP SIGN, 1ST FRACTION
2988    110F  4E                      MOV   C,M       ;OPERAND SIGN AND 1ST FRACTION
2989    1110  2C                INX   INR   L         ;TO ADDRESS OPERAND 2ND FRACTION
2990    1111  56                      MOV   D,M       ;OPERAND 2ND FRACTION
2991    1112  2C                INX   INR   L         ;TO ADDRESS OPERAND 3RD FRACTION
2992    1113  5E                      MOV   E,M       ;OPERAND 3RD FRACTION
2993    1114  26 00                   MVI   H,SCRB    ;TO ADDRESS SCRATCH BANK
2994    1116  2E 35                   MVI   L,ACCE    ;TO ADDRESS ACCUM EXPOENT
2995    1118  7E                      MOV   A,M       ;ACCUM EXPONENT
2996    1119  A7                      ANA   A         ;SET CONTROL BITS
2997    111A  C8                      RZ              ;RETURN IF ACCUM ZERO
2998    111B  80                      ADD   B         ;RESULT EXPONENT PLUS BIAS
2999    111C  47                      MOV   B,A       ;RESULT EXPONENT PLUS BIAS
3000    111D  1F                      RAR             ;CARRY TO SGN
3001    111E  A8                      XRA   B         ;CARRY AND SIGN MUST DIFFER
3002    111F  78                      MOV   A,B       ;RESULT EXPONENT PLUS BIAS
3003    1120  06 80                   MVI   B,2000    ;EXP BIAS SIGN MASK MS BIT
3004    1122  F2 30 11                JP    OVUN      ;IF OVERFLOW OR UNDERFLOW
3005    1125  90                      SUB   B         ;REMOVE EXESS EXP BIAS
3006    1126  C8                      RZ              ;RETURN IF UNDERFLOW
3007    1127  77                      MOV   M,A       ;RESULT EXPONENT
3008    1128  2C                      INR   L         ;TO ADDRESS ACCUMULATOR SIGN
3009    1129  7E                      MOV   A,M       ;ACCUM SIGN
3010    112A  A9                      XRA   C         ;RESULT SIGN IN SIGN BIT
3011    112B  A0                      ANA   B         ;RESULT SIGN
3012    112C  77                      MOV   M,A       ;RESULT SIGN
3013    112D  79                      MOV   A,C       ;OPERAND SIGN AND FIRST FRACTION
3014    112E  B0                      ORA   B         ;OPERAND 1ST FRACTION
3015    112F  C9                      RET             ;RETURN TO CALLER
3016    1130  07              OVUN:   RLC   ;         ;SET CARRY BIT IF OVERFLOW
3017    1131  D8                      RC              ;RETURN IF OVERFLOW
3018    1132  AF                      XRA   A         ;ZERO
3019    1133  C9                      RET             ;RETURN IF UNDERFLOW
3020                          ;     SUBROUTINE TO LEFT SHIFT THE B, C,
3021                          ;     D, AND E REGISTERS ONE BIT
3022    1134  7B              LSH:    MOV   A,E       ;ORIGINAL CONTENTS OF E
3023    1135  17                      RAL             ;LEFT SHIFT E
3024    1136  5F                      MOV   E,A       ;RESTORE CONTENTS OF E REGISTER
3025    1137  7A              LSH1:   MOV   A,D       ;ORIGINAL CONTNTS OF D REGISTER
3026    1138  17                      RAL   ;         ;LEFT SHIFT D
3027    1139  57                      MOV   D,A       ;RESTORE CONTENTS OF D REGISTER
3028    113A  79                      MOV   A,C       ;ORIGINAL CONTENTS OF C REGISTER
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

3029     113B   17                        RAL             ;LEFT SHIFT C
3030     113C   4F                        MOV   C,A       ;RESTORE CONTENTS OF C REGISTER
3031     113D   78                        MOV   A,B       ;ORIGINAL CONTENTS OF B REGISTER
3032     113E   8F                        ADC   A         ;LEFT SHIFT B
3033     113F   47                        MOV   B,A       ;RESTORE CONTENTS OF B REGISTER
3034     1140   C9                        RET             ;RETURN TO CALLER
3035                             ;     RIGHT SHIFT THE B, C, D, AND E REGISTER
3036                             ;     BY THE SHIFT COUNT IN THE A REGISTER
3037                             ;     SHIFT OPERAND TO REGISTER INDICATED BY
3038                             ;     SHIFT COUNT
3039     1141   1E 00             RSH:      MVI   E,0       ;OPERAND 4TH FRACTION IS ZERO
3040     1143   2E 08             RSH0:     MVI   L,0100    ;EACH REG IS 8 BITS OF SHIFT
3041     1145   BD                RSH1:     CMP   L         ;COMPARE SHIFT COUNT TO 8
3042     1146   FA 52 11                    JM    RSH2      ;IF REQ SHIFT LESS THAN 8
3043     1149   5A                        MOV   E,D       ;OPERAND 4TH FRACTION
3044     114A   51                        MOV   D,C       ;OPERAND 3RD FRACTION
3045     114B   48                        MOV   C,B       ;OPERAND 2ND FRACTION
3046     114C   06 00                     MVI   B,0       ;OPERAND 1ST FRACTION IS ZRRO
3047     114E   95                        SUB   L         ;REDUCE SHIFT COUNT BY 1 REFG
3048     114F   C2 45 11                  JNZ   RSH1      ;IF MORE SHIFTS REWQUIRED
3049                             ;     SHIFT OPERAND RIGHT BY -SHIFT COUNT-
3050                             ;     BITS.
3051     1152   A7                RSH2:     ANA   A         ;SET CONTROL BITS
3052     1153   C8                        RZ              ;RETURN IF SHIFT COMPLETE
3053     1154   6F                        MOV   L,A       ;SHIFT COUNT
3054     1155   A7                RSH3:     ANA   A         ;CLEAR CARRY BIT
3055     1156   78                        MOV   A,B       ;OPERAND 1ST FRACTION
3056     1157   1F                        RAR             ;RIGHT SHIFT OP 1ST FRACTION
3057     1158   47                        MOV   B,A       ;OPERAND 1ST FRACTION
3058     1159   79                        MOV   A,C       ;OPERAND 2ND FRACTION
3059     115A   1F                        RAR             ;FRIGHT SHIFT OP 2ND FRACTOIN
3060     115B   4F                        MOV   C,A       ;OPERAND 2ND FRACTION
3061     115C   7A                        MOV   A,D       ;OPERAND 3RD FRACTION
3062     115D   1F                        RAR             ;RIGHT SHIFT OP 3RD FRACTION
3063     115E   57                        MOV   D,A       ;OPERAND 3RD FRACTOON
3064     115F   7B                        MOV   A,E       ;OPERAND 4TH FRACTION
3065     1160   1F                        RAR             ;RIGHT SHIFT OP 4TH FRACTION
3066     1161   5F                        MOV   E,A       ;OPERAND 4TH FRACTION
3067     1162   2D                        DCR   L         ;DECREMENT SHIFT COUNT
3068     1163   C2 55 11                  JNZ   RSH3      ;IF MORE SHIFTS REQUIRED
3069     1166   C9                        RET             ;RETURN TO CALLER
3070                             ;     COMPLEMENT THE B, C, D, AND E REGISTERS
3071     1167   2D          DCX H COMP:   DCR   L         ;TO ADDR ACCUM SIGN
3072     1168   7E                        MOV   A,M       ;ACCUMALUTAOR SIGN
3073     1169   EE 80                     XRI   2000      ;CHANGE SIGN
3074     116B   77                        MOV   M,A       ;ACCUMALATOR SIGN
3075     116C   AF                COMP1:    XRA   A         ;ZERO
3076     116D   6F                        MOV   L,A       ;ZERO
3077     116E   93                        SUB   E         ;COMPLEMENT 4TH FRACTION
3078     116F   5F                        MOV   E,A       ;4TH FRACTION
3079     1170   7D                        MOV   A,L       ;ZERO
3080     1171   9A                        SBB   D         ;COMPLEMENT 3RD FAACTION
3081     1172   57                        MOV   D,A       ;3RD FRACTION
3082     1173   7D                        MOV   A,L       ;ZERO
3083     1174   99                        SBB   C         ;COMPLEMENT 2ND VFRACTION
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

3084    1175  4F                        MOV     C,A       ;2ND FRACTION
3085    1176  7D                        MOV     A,L       ;ZERO
3086    1177  98                        SBB     B         ;COMPLEMENT 1ST FRACTION
3087    1178  47                        MOV     B,A       ;1ST FRACTION
3088    1179  C9                        RET               ;RETURN TO CALLER
3089                        ;     NORMALIZE THE REGISTERS
3090    117A  2E 20         NORM:     MVI     L,0400    ;MAX NORMALIZING SHIFT
3091    117C  78            NORM1:    MOV     A,B       ;1ST FRACTION
3092    117D  A7                      ANA     A         ;SET CONTROL BITS
3093    117E  C2 9A 11                JNZ     NORM3     ;IF 1ST FRACTION NONZERO
3094    1181  41                      MOV     B,C       ;1ST FRACTION
3095    1182  4A                      MOV     C,D       ;2ND FRACTION
3096    1183  53                      MOV     D,E       ;3RD FRACTION
3097    1184  5F                      MOV     E,A       ;ZEROTH 4TH FRACTIOLN
3098    1185  7D                      MOV     A,L       ;NORMALIZING SHIFT COUNT
3099    1186  D6 08                   SUI     0100      ;REDUCE SHIFT COUNT
3100    1188  6F                      MOV     L,A       ;NORMALIZING SHIFT COUNT
3101    1189  C2 7C 11                JNZ     NORM1     ;IF FRACTIOLN NONZERO
3102    118C  C9                      RET               ;IF FRATION ZERO
3103    118D  2D            NORM2:    DCR     L         ;DECREMENT SHIFT COUNT
3104    118E  7B                      MOV     A,E       ;ORIGINAL CONTENTS OF E
3105    118F  17                      RAL               ;LEFT SHIFT E
3106    1190  5F                      MOV     E,A       ;RESTORE THE CONTENTS OF REGISTER E
3107    1191  7A                      MOV     A,D       ;ORIGINAL CONTENTS OD D REGISTER
3108    1192  17                      RAL               ;LEFT SHIFT D
3109    1193  57                      MOV     D,A       ;RESTORF CONTENTS OF D REGISTER
3110    1194  79                      MOV     A,C       ;ORIGINAL CONTENTS OF C REGISTER
3111    1195  17                      RAL               ;LEFT SHIFT C
3112    1196  4F                      MOV     C,A       ;RESTORE THE CONTENTS OF C REGISTER
3113    1197  78                      MOV     A,B       ;ORIGINAL CONTENTS OF REGISTER B
3114    1198  8F                      ADC     A         ;LEFT SHIFT B
3115    1199  47                      MOV     B,A       ;RESTORE CONTENTS OF B REGISTER
3116    119A  F2 8D 11      NORM3:    JP      NORM2     ;IF NOT NORMALIZED
3117    119D  7D                      MOV     A,L       ;NORMALIZING SHIFT COUNT
3118    119E  D6 20                   SUI     0400      ;REMOVE BIAS
3119    11A0  2E 35                   MVI     L,ACCE    ;TO ADDR ACCUM EXPONENT
3120    11A2  86                      ADD     M         ;ADJUST ACCUM EXPONENT
3121    11A3  77                      MOV     M,A       ;NEW ACCUM FXPONENT
3122    11A4  C8                      RZ                ;RETURN IF ZERO EXPONENT
3123    11A5  1F                      RAR               ;BORROW BIT TO SIGN
3124    11A6  A7                      ANA     A         ;SET SIGN TO IND. UNDERFLOW
3125    11A7  C9                      RET               ;RETURN TO CALLER
3126                        ;     SUBROUTINE TO ROUND THE B, C, D REGISTERS
3127    11A8  2E 35         ROND:     MVI     L,ACCE    ;TO ADDR ACCUM EXPONENT
3128    11AA  7B                      MOV     A,E       ;4TH FRACTION
3129    11AB  A7                      ANA     A         ;SET CONTROL BITS
3130    11AC  5E                      MOV     E,M       ;ACCUM EXPONENT
3131    11AD  FC B7 11                CM      RNDR      ;CALL2ND LEVEL ROUNDFR
3132    11B0  D8                      RC                ;IF OBERFLOW
3133    11B1  78                      MOV     A,B       ;1ST FRACTION
3134    11B2  2C                      INR     L         ;TO ADDR ACCUM SGN
3135    11B3  AE                      XRA     M         ;ACCUM SIGN AND 1ST FRACTION
3136    11B4  C3 B7 0F                JMP     STR1      ;RETURN THRU STORE SUBR.
3137                        ;     SECOND LEVEL ROUNDING SUBROUTINF
3138    11B7  14            RNDR:     INR     D         ;ROUND 3RD FRACTION
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

3139     11B8   C0                       RNZ             ;RETURN IF NO CARRY
3140     11B9   0C                       INR     C       ;CARRY TO 2ND FRACTION
3141     11BA   C0                       RNZ             ;RETURN IF NO CARRY
3142     11BB   04                       INR     B       ;CARRY TO 1ST FRACTION
3143     11BC   C0                       RNZ             ;RETURN IF NO CARRY
3144     11BD   7B                       MOV     A,E     ;ACCUM EXPOENT
3145     11BE   C6  01                   ADI     1       ;INCREMENT ACCUM EXPONENT
3146     11C0   5F                       MOV     E,A     ;NEW ACCUM EXPONENT
3147     11C1   06  80                   MVI     B,2000  ;NEW 1ST FRACTION
3148     11C3   77                       MOV     M,A     ;NEW ACCUM EXPONENT
3149     11C4   C9                       RET             ;RETURN TO ROND SUBROUTINE
3150                             ;       FIXED POINT MULTIPLY SUBROUTINE
3151     11C5   2E  0E           MULX:   MVI     L,MULP1 ;TO ADDR 1ST MULTIPLICAND
3152     11C7   77                       MOV     M,A     ;1ST MULTIPLICAND
3153     11C8   2E  0A                   MVI     L,MULP2 ;TO ADDR 2ND MULTIPLICAND
3154     11CA   72                       MOV     M,D     ;2ND MULTIPLICAND
3155     11CB   2E  06                   MVI     L,MULP3 ;TO ADDR 3RD MULTIPLICAND
3156     11CD   73                       MOV     M,E     ;R3RD MULTIPLICAND
3157     11CE   AF                       XRA     A       ;CLEAR 6TH PRODUCT
3158     11CF   5F                       MOV     E,A     ;CLEAR 5TH PRODUCT
3159     11D0   57                       MOV     D,A     ;CLEAR 4TH PRODUCT
3160                             ;       MULTIPLY BY EACH ACCUMULATOR
3161                             ;       FRACTION IN TURN
3162     11D1   2E  39                   MVI     L,ACC3  ;TO ADDR 3RD FRACTION
3163     11D3   CD  E0  11               CALL    MULX2   ;MULTIPLY BY ACCUM 3RD FRACTION
3164     11D6   2E  38                   MVI     L,ACC2  ;TO ADDR 2ND FRACTION
3165     11D8   CD  DD  11               CALL    MULX1   ;MULTIPLY BY ACCUM 2ND FRACTION
3166     11DB   2E  37                   MVI     L,ACC1  ;TO ADDR 1ST FRACTION
3167                             ;       MULTIPLY BY ONE ACCUMULATOR WORD
3168     11DD   7A               MULX1:  MOV     A,D     ;5TH PARTIAL PRODUCT
3169     11DE   59                       MOV     E,C     ;4TH PARTIAL PRODUVT
3170     11DF   50                       MOV     D,B     ;3RD PARTIAL PRODUVT
3171     11E0   46               MULX2:  MOV     B,M     ;MULTIPLIER
3172     11E1   6F                       MOV     L,A     ;5TH PARTIAL PRODUCT
3173     11E2   AF                       XRA     A       ;ZERO
3174     11E3   4F                       MOV     C,A     ;2ND PARTIAL PRODUCT
3175     11E4   90                       SUB     B       ;SET CARRY BIT FOR EXIT FLAG
3176     11E5   DA  F1  11               JC      MULX3   ;IF MULTIPLIER IS NOT ZERO
3177     11E8   4A                       MOV     C,D     ;2ND PARTIAL PRODUCT
3178     11E9   53                       MOV     D,E     ;3RD PARTIAL PRODUCT
3179     11EA   C9                       RET             ;MULT BY ZERO COMPLETE
3180                             ;       COMPLETE ADDITION OF MULTIPLICAND.
3181     11EB   4F               MULX5:  MOV     C,A     ;2ND PARTIAL PRODUCT
3182     11EC   D2  F1  11               JNC     MULX3   ;IF NO CARRY TO 1ST PRODUCT
3183     11EF   04                       INR     B       ;ADD CARRY TO 1ST PRODUCT
3184     11F0   A7                       ANA     A       ;CLEAR CARRY BIT
3185                             ;       LOOP FOR EACH BIT OF MULTIPLIER WORD.
3186     11F1   7D               MULX3:  MOV     A,L     ;5TH PARTIAL PRODUCT, EXIT FLAG
3187     11F2   8F                       ADC     A       ;SHIFT EXIT FLAG OUT IF DONE
3188     11F3   C8                       RZ              ;EXIT IF MULTIPLICATION DONE
3189     11F4   6F                       MOV     L,A     ;5TH PART PRODUCT, EXIT FLAG
3190     11F5   7B                       MOV     A,E     ;4TH PARTIAL PRODUCT
3191     11F6   17                       RAL             ;SHIFT 4TH PARTIAL PRODUCT
3192     11F7   5F                       MOV     E,A     ;4TH PARTIAL PRODUCT
3193     11F8   7A                       MOV     A,D     ;3RD PARTIAL PRODUCT
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

3194     11F9   17                        RAL                 ;SHIFT 3RD PARTIOL PRODUCT
3195     11FA   57                        MOV     D,A         ;3RD PARTIAL PRODUCT
3196     11FB   79                        MOV     A,C         2NDD PARTIAL PRODUCT
3197     11FC   17                        RAL                 ;SHIFT 2ND PARTIAL PRODUCT
3198     11FD   4F                        MOV     C,A         ;2ND PARTIAL PRODUCT
3199     11FE   78                        MOV     A,B         ;1ST PARTIAL PRODUCT AND MULTIPLIER
3200     11FF   17                        RAL                 ;SHIFT 1ST PARTIAL PRODUCT AND MULTIPLIER
3201     1200   47                        MOV     B,A         ;1ST PARTIAL PRODUCT AND MULTIPLIER
3202     1201   D2  F1  11                JNC     MULX3       ;IF NO ADDITION REQUIRED
3203                              ;       ADD THE MULTIPLICAND TO THE PRODUCT
3204                              ;       IF THE MULTIPLIER BIT IS ONE.
3205     1204   7B                        MOV     A,E         ;4TH PARTIAL PRODUCT
3206     1205   C3  05  00                JMP     MULX4       ;TO RAM CODE
3207                              ;       FIXED POINT DIVIDE SUBROUTINE
3208                              ;       SUBTRACT DIVISOR FROM ACCUMULATOR TO
3209                              ;       OBTAIN 1ST REMAINDER
3210     1208   2E  39            DIVX:   MVI     L,ACC3      ;TO ADDRESS ACCUM 3RD FRACTION
3211     120A   7E                        MOV     A,M         ;ACCUMALTOR 3RD FRACTION
3212     120B   93                        SUB     E           ;DIVISOR 3RD FRACTION
3213     120C   77                        MOV     M,A         ;REMAINDER 3RD FRACTION
3214     120D   2D                        DCR     L           ;TO ADDRESS ACCUM 2ND FRACTION
3215     120E   7E                        MOV     A,M         ;ACCUMALOTOR 2ND FRACTIN
3216     120F   9A                        SBB     D           ;DIVISOR 2ND FRACTION
3217     1210   77                        MOV     M,A         ;REMAINDER 2ND FRACTION
3218     1211   2D                        DCR     L           ;TO ADDRESS ACCUM 1ST FRACTION
3219     1212   7E                        MOV     A,M         ;ACCUMULATOR 1ST FRACTION
3220     1213   99                        SBB     C           ;DIVISOR 1ST FRACTION
3221     1214   77                        MOV     M,A         ;REMAINDER 1ST FRACTION
3222                              ;       HALVE THE DIVISOR AND STORE FFOR
3223                              ;       ADDITION OR SUBRRACTION
3224     1215   79                        MOV     A,C         ;DIVISOR 1ST FRACTION
3225     1216   17                        RAL                 ;SET ACRRY BIT
3226     1217   79                        MOV     A,C         ;DIVISOR 1ST FRACTION
3227     1218   1F                        RAR                 ;HALF DIVISOR 1ST FRACTION
3228                              ;       +200B TO COORRECT QUOTIENT
3229     1219   2E  1E                    MVI     L,OP1S      ;TO ADDR 1ST SUBTRACT DIVISOR
3230     121B   77                        MOV     M,A         ;1ST SUBRRACT DIVISOR
3231     121C   2E  2C                    MVI     L,OP1A      ;TO ADDR 1ST ADD DIVISOR
3232     121E   77                        MOV     M,A         ;1ST ADD DIVISOR
3233     121F   7A                        MOV     A,D         ;DIVISOR 2ND FRACTION
3234     1220   1F                        RAR                 ;HALF OF DIVISOR 2ND DIGIT
3235     1221   2E  1A                    MVI     L,OP2S      ;TO ADDR 2ND SUBTAACT DIVISOR
3236     1223   77                        MOV     M,A         ;2ND SUBTRACT DIVISRO
3237     1224   2E  28                    MVI     L,OP2A      ;TO ADDR 2ND ADD DIVISOR
3238     1226   77                        MOV     M,A         ;2ND ADD DIVIOR
3239     1227   7B                        MOV     A,E         ;DIVISOR 3RD FRACTION
3240     1228   1F                        RAR                 ;JALF OF DIVISOR 3RD FRACTION
3241     1229   2E  16                    MVI     L,OP3S      ;TO ADDRESS 3RD SUBRRACT DIVISOR
3242     122B   77                        MOV     M,A         ;3RD SUBTRACT DIVISOR
3243     122C   2E  24                    MVI     L,OP3A      ;TO ADDRESS 3RD ADD DIVISOR
3244     122E   77                        MOV     M,A         ;3RD ADD DIVISOR
3245     122F   06  00                    MVI     B,0         ;INIT QUOTIENT 1ST FRACTION
3246     1231   78                        MOV     A,B         ;DIVISOR FOURTH FRACTION 2S ZRRO
3247     1232   1F                        RAR                 ;LOW BIT OF DIVISOR 3RD FRACTION
3248     1233   2E  13                    MVI     L,OP4S      ;TO ADDRESS 4TH SUBRRACT DIVISOR
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

3249     1235   77                       MOV     M,A       ;4TH SUBTRACT DIVISOR
3250     1236   2E  21                   MVI     L,OP4A    ;TO ADDRESS 4TH ADD DIVISOR
3251     1238   77                       MOV     M,A       ;4TH ADD DIVISOR
3252     1239   2E  2F                   MVI     L,OP4X    ;TO ADDRESS 4TH ADD DIVISOR
3253     123B   77                       MOV     M,A       ;4TH ADD DIVISOR
3254                             ;     LOAD 1ST REMAINDER, CHECK SIGN
3255     123C   2E  37                   MVI     L,ACC1    ;TO ADDR REMAINDER 1ST FRACTION
3256     123E   7E                       MOV     A,M       ;REMAINDER 1ST FRACTION
3257     123F   2C                       INR     L         ;TO ADDR REMAINDER 2ND FRACTION
3258     1240   56                       MOV     D,M       ;REMAINDER 2ND FRACTION
3259     1241   2C                       INR L   ;TO ADDR REMAINDER 3RD FRACTION
3260     1242   5E                       MOV     E,M       ;REMAINDER 3RD FRACTION
3261     1243   A7                       ANA     A         ;SET CONTROL BITS
3262     1244   FA  6E  12               JM      DIVX4     ;IF REMAINDER IS BNEGATIVE
3263                             ;     ADJUST EXPONENT, POSITION REMAINDER
3264                             ;     AND INITIALIZE THE QUOTIENT
3265     1247   2E  35                   MVI     L,ACCE    ;TO ADDRESS ACCUM EXPONENT
3266     1249   4E                       MOV     C,M       ;QUOTIENT EXPONENT
3267     124A   0C                       INR     C         ;INCREMENT QUOTIENT EXPONENT
3268     124B   C8                       RZ                ;RETURN IF OVERVLOW
3269     124C   71                       MOV     M,C       ;QUOTIENT EXPONENT
3270     124D   6B                       MOV     L,E       ;REMEINDER 3RD FRACTION
3271     124E   62                       MOV     H,D       ;REMAINDER 2ND FRACTION
3272     124F   5F                       MOV     E,A       ;REMAINDER 1ST FRACTION
3273     1250   16  01                   MVI     D,1       ;INITIALIZE QUOTIENT 3RD FRACTION
3274     1252   48                       MOV     C,B       ;INITIALIZE QUOTIENT 2ND FRACTION
3275                             ;     SUBTRACT THE DIVISOR FROM THE REMAINDER
3276                             ;     IF IT IS POSITIVE
3277     1253   AF              DIVX1:   XRA     A         ;REMAINDER 4TH FRACTION IS ZERO
3278     1254   CD  12  00               CALL    DIVX5     ;CALL RAM SECTION
3279     1257   07              DIVX2:   RLC               ;SHIFT REM 4TH FRACTION TO CY
3280                             ;     SHIFT THE REMAINDER LEFT ONE BIT
3281     1258   78                       MOV     A,B       ;QUOTIENT 1ST FRACTION
3282     1259   17                       RAL               ;MX BIT OF QUOTIENT TO CY
3283     125A   D8                       RC                ;IF DIVISION COMPLETE
3284     125B   1F                       RAR               ;REMAINDER 4TH FRACTION TO CY
3285     125C   7D                       MOV     A,L       ;REMAINDER 3RD F*AZCTION
3286     125D   17                       RAL               ;LEFT SHIFT REM 3RD FRACTION
3287     125E   6F                       MOV     L,A       ;REMAINDER 3RD FRACTION
3288     125F   7C                       MOV     A,H       ;REMAINDER 2ND FRACTION
3289     1260   17                       RAL               ;LEFT SHIFT REM 2ND FRACTIN
3290     1261   67                       MOV     H,A       ;REMAINDER 2ND FRACTION
3291     1262   CD  34  11               CALL    LSH       ;CALL LEFT SHIFT ROUTINE
3292                             ;     BRANCH IF SUBTARACTION IS REQUIRD
3293     1265   7A                       MOV     A,D       ;QUOTIENT 3RD REMAINDER
3294     1266   0F                       RRC               ;REM SIGN INDIC TO CARRY BIT
3295     1267   DA  53  12               JC      DIVX1     ;TO SUB DIVISOR IF REM POS
3296                             ;     ADD THE DIVISOR IF THE REMAINDER
3297                             ;     IS BNEGATIVE
3298     126A   7D              DIVX3:   MOV     A,L       ;REMAINDER 3RD FRACTION
3299     126B   C3  23  00               JMP     DIVX6     ;TO RAM CODE
3300                             ;     POSITION THE REMAINDER AND INITIALIZE
3301                             ;     THE QUOTIENT
3302     126E   6B              DIVX4:   MOV     L,E       ;REMAINDER 3RD FRACTION
3303     126F   62                       MOV     H,D       ;REMAINDER 2ND FRACTION
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT
3304    1270  5F                        MOV     E,A       ;REMAINDER 1ST FRACTION
3305    1271  50                        MOV     D,B       ;INITIALIZE QUOTIENT 3RD FRACTION
3306    1272  48                        MOV     C,B       ;INITIALIZE QOUTIENT REMAINDER 2ND FRA TION
3307    1273  C3 6A 12                  JMP     DIVX3     ;ADD DIVISOR IF REM IS NEG.
3308    1276  00                        DB      0         ;CHECKSUM
3309
3310                              ;     8008 BINARY FLOATING POINT SYSTEM
3311                              ;     FORMAT CONVERSION PACKAGE
3312                              ;     PROGRAMMER: CAL OHME
3313                              ;     DATE 26 DECEMBER 1973
3314                              ;     ARITH IS THE BEFINNING ADDRESS OF
3315                              ;     ARITHMETIC AND UTILITY PACKAGE OF THE FLOATING
3316                              ;     POINT SYSTEM
3317                              ;     SCR IS THE BEGINNING OF THE
3318                              ;     RAM USED AS SCRATCHPAD FOR THE SYSTEM.
3319                              ;     RAM LOCATIONS USED BY THE BINARY
3320                              ;     FLOATING POINT SYSTEM
3321    003B                      ADRL:     EQU     SF+1      ;CHARACTER SRTRING WORD
3322    003C                      ADRH:     EQU     ADRL+1    ;CHARACTER STRING BANK
3323    003D                      TMP1:     EQU     ADRH+1    ;;TEMP STORAGE
3324    003E                      TMP2:     EQU     TMP1+1    ;TEMMP STORAGE
3325    003F                      TMP3:     EQU     TMP2+1    ;TEMMP STORAGE
3326    0040                      VALE:     EQU     TMP3+1    ;VALUE EXPONENT
3327    0041                      VAL1      EQU     VALE+1    ;VALUE 1ST FRACTION
3328    0042                      VAL2      EQU     VAL1+1    ;VALUE 2ND FRACTION
3329    0043                      VAL3      EQU     VAL2+1    ;VALUE 3RD FRACTION
3330    0044                      TMP4      EQU     VAL3+1    ;TEMP STORAGE
3331                              ;     SUBROUTINE TO CONVERT FROM FIXED
3332                              ;     POINT TO FLOATING POINT FORMAT.
3333    1277  6B                  FLT:      MOV     L,E       ;INPUT EXPONENT
3334    1278  5A                            MOV     E,D       ;4TH INPUT FRACTION
3335    1279  51                            MOV     D,C       ;3RD INPUT FRACTIN
3336    127A  48                            MOV     C,B       ;2ND INPUT FRACTION
3337    127B  47                            MOV     B,A       ;1ST INPUT FRACTION
3338    127C  7D                            MOV     A,L       ;INPUT EXPONENT
3339    127D  EE 80                         XRI     2000      ;APPLY EXPONENT BIAS
3340    127F  26 00                         MVI     H,SCRB    ;TO ADDRESS SCRATCH BANK
3341    1281  2E 35                         MVI     L,ACCE    ;TO ADDR ACCUM EXPONENT
3342    1283  77                            MOV     M,A       ;ACCUM EXPONENT
3343    1284  2C                            INR     L         ;TO ADDRESS ACCUM SIGN
3344    1285  36 80                         MVI     M,2000    ;SET ACCUM SIGN POSITIVE
3345    1287  2C                            INR     L         ;TO ADDR ACCUM 1ST FRACTION
3346    1288  78                            MOV     A,B       ;1ST INPUT FRACTION
3347    1289  A7                            ANA     A         ;SET SIGN BIT
3348    128A  17                            RAL               ;INPUT SIGN TO CARRY
3349    128B  C3 E3 10                      JMP     ADD10     ;COMPLETE CONVERSION
3350                              ;     SUBROUTINE TO CONVERT FROM FLOATING
3351                              ;     POINT TO FISEX POINT FORMAT
3352    128E  26 00               FIX:      MVI     H,SCRB    ;TO ADDRESS SCRATCH BANK
3353    1290  2E 35                         MVI     L,ACCE    ;TO ADDR ACCUM EXPONENT
3354    1292  7E                            MOV     A,M       ;ACCUM EXPONENT
3355    1293  A7                            ANA     A         ;SET CONTROL BITS
3356    1294  CA BC 12                      JZ      FIX1      ;IF ACCUM IS SET TO ZERO
3357    1297  7B                            MOV     A,E       ;INPUT EXPONENT
3358    1298  C6 7F                         ADI     1770      ;APPLY BIAS-1
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

3359    129A  96                      SUB     M          ;SHIFT COUNT=1
3360    129B  D8                      RC                 ;RETURN IF ACCUM TOO LARGE
3361    129C  FE 1F                   CPI     0370       ;COMPARE TO LARGE SHIFT
3362    129E  D2 BC 12                JNC     FIX1       ;IF ACCUM TOO SMALL
3363    12A1  C6 01                   ADI     1          ;SHIFT COUNT
3364    12A3  2E 37                   MVI     L,ACC1     ;TO ADDR ACCUM 1ST FRACTION
3365    12A5  46                      MOV     B,M        ;ACCUM 1ST FRACTION
3366    12A6  2C                      INR     L          ;TO ADDR ACCUM 2ND FRACTION
3367    12A7  4E                      MOV     C,M        ;ACCUM 2ND FRACTION
3368    12A8  2C                      INR     L          ;TO ADDR ACCUM 3RD FRACTION
3369    12A9  56                      MOV     D,M        ;ACCUM 3RD FRACTOON
3370    12AA  CD 41 11                CALL    RSH        ;POSITION THE FRACTION
3371    12AD  2E 36                   MVI     L,ACCS     ;TO ADDR ACCUM SIGN
3372    12AF  7E                      MOV     A,M        ;ACCUM SIGN
3373    12B0  A7                      ANA     A          ;SET CONTROL BITS
3374    12B1  F4 67 11                CP      COMP       ;COMPLEMENT FRACTION IF NEG
3375    12B4  3E 01                   MVI     A,1        ;NON-ZERO
3376    12B6  B0                      ORA     B          ;SET CONTROL BITS FOR EXIT
3377    12B7  78                      MOV     A,B        ;UST REAULT
3378    12B8  41                      MOV     B,C        ;2ND REAULT
3379    12B9  4A                      MOV     C,D        ;3RD REAULT
3380    12BA  53                      MOV     D,E        ;4TH RESULT
3381    12BB  C9                      RET                ;RETURN TO CALLER
3382    12BC  AF              FIX1:   XRA     A          ;ZERO
3383    12BD  47                      MOV     B,A        ;ZERO
3384    12BE  4F                      MOV     C,A        ;ZERO
3385    12BF  57                      MOV     D,A        ;ZERO
3386    12C0  C9                      RET                ;RETURN TO CALLER
3387    12C1        00                DB      0          ;CHECKSUM WORD
3388                  ;       INP SUBROUTINE ENTRY POINT
3389                  ;       INITIALIZE TEMPORARY STORAGE
3390    12C2  5E              INP:    MOV     E,M        ;FIRST CHARACTER OF STRINT
3391    12C3  CD 4C 14                CALL    SVAD       ;SET CHAR ADDR, PNT FLAG, EXP
3392    12C6  2C                      INR     L          ;TO ADDRESS VALUE SIGN
3393    12C7  36 80                   MVI     M,2000     ;SET VALUE SIGN POSITIVE
3394    12C9  2E 35                   MVI     L,ACCE     ;TO ADDR ACCUM EXPONENT
3395    12CB  72                      MOV     M,D        ;SET ACCUM TO ZERO
3396    12CC  7B                      MOV     A,E        ;FIRST CAHRACTER
3397    12CD  FE F0                   CPI     3600       ;COMPARE TO SPACE
3398    12CF  CA DF 12                JZ      INP1       ;IF SPACE CHARACTER
3399    12D2  FE FB                   CPI     3730       ;COMPARE CHAR TO PLUS
3400    12D4  CA DF 12                JZ      INP1       ;IF PLUS SIGN
3401    12D7  FE FD                   CPI     3750       ;COMPARE TO MINUS
3402    12D9  C2 E5 12                JNZ     INP2       ;IF NOT MINUS
3403    12DC  2E 3F                   MVI     L,TMP3     ;TO ADDR VALUE SIGN
3404    12DE  72                      MOV     M,D        ;SET VALUE SIGN NEGATIVE
3405                  ;       ANALYZE NEXT CHARACTER IN STRING.
3406    12DF  CD 59 14        INP1:   CALL    CHAD       ;CALL CHAR ADDR SBRTN
3407    12E2  7E                      MOV     A,M        ;NEXT CHARACTER
3408    12E3  26 00                   MVI     H,SCRB     ;TO ADDRESS SCRATCH BANK
3409    12E5  06 00           INP2:   MVI     B,0        ;DIGIT 2ND WD OR DEC EXP
3410    12E7  FE FE                   CPI     3760       ;COMPARE TO DECIMAL POINT
3411    12E9  CA 22 13                JZ      INP3       ;INF DECIMAL POINT
3412    12EC  FE 15                   CPI     0250       ;COMPARE TO EXPONENT SIGN
3413    12EE  CA 2C 13                JZ      INP4       ;IF EXPONENT SIGN
```

```
STMNT     LOC     OBJECT CODE       SOURCE STATEMENT

3414      12F1    FE  0A                    CPI     0120      ;SET CARRY IF CHAR IS DIGIT
3415      12F3    D2  5D  13                JNC     INP8      ;IF CHAR IS NOT A DIGIT
3416      12F6    2E  44                    MVI     L,TMP4    ;TO ADDR CURRENT DIGIT
3417      12F8    77                        MOV     M,A       ;SAVE CURRENT DIGIT
3418      12F9    21  64  14                LXI     H,FTEN    ;TO ADDR FLOATING SIGN
3419      12FC    CD  04  10                CALL    MUL       ;MULTIPLY BY TEN
3420      12FF    2E  40                    MVI     L,VALE    ;TO ADDR VALUE
3421      1301    CD  B6  0F                CALL    STR       ;STORE OLD VALUES TIMES TEN
3422      1304    2C                        INR     L         ;TO ADDR CURRENT DIGIT
3423      1305    7E                        MOV     A,M       ;CURRENT DIGIT
3424      1306    06  00                    MVI     B,0       ;CLEAR 2ND WORD OF DIGIT
3425      1308    48                        MOV     C,B       ;CLEAR WORD OF DIGIT
3426      1309    50                        MOV     D,B       CLEAR 4THRD OF DIGIT
3427      130A    1E  08                    MVI     E,0100    ;INDICATE DIGIT IS IN REG A
3428      130C    CD  77  12                CALL    FLT       ;CONVERT DIGIT TO FLOATING PNT
3429      130F    2E  40                    MVI     L,VALE
3430      1311    CD  4F  10                CALL    AD        ;ADD OLD VALUE TIMES TEN
3431      1314    2E  3E                    MVI     L,TMP2    ;TO ADDR DEC PNT FLAG
3432      1316    7E                        MOV     A,M       ;DECIMAL POINT FLAG
3433      1317    A7                        ANA     A         ;SET CONTROL BITS
3434      1318    CA  DF  12                JZ      INP1      ;IF NO DEC PNT ECNCOUNTERED
3435      131B    2D                        DCR     L         ;TO ADDR INPUT EXPONENT
3436      131C    46                        MOV     B,M       ;INPUT EXPONENT
3437      131D    05                        DCR     B         ;DECREMENT INPUT EXPONENT
3438      131E    70                        MOV     M,B       ;UPDATE INPUT EXPONENT
3439      131F    C3  DF  12                JMP     INP1      ;TO GET NEXT CHARACTER
3440      1322    2E  3E          INP3:     MVI     L,TMP2    ;TO ADDR DEC PNT GFLAG
3441      1324    AE                        XRA     M         ;ZERO IF FLAG SET
3442      1325    77                        MOV     M,A       ;SET DEC PNT FLAG
3443      1326    C2  DF  12                JNZ     INP1      ;IF LFLAG NOT ALREADY THERE
3444      1329    C3  5D  13                JMP     INP8      ;IF 2ND DEC PT
3445                              ;     PROCESS DECIMAL EXPONENT
3446      132C    CD  59  14      INP4:     CALL    CHAD      ;CALL CHAR ADDR SBRTN
3447      132F    7E                        MOV     A,M       ;NEXT CHARACTER OF STRING
3448      1330    47                        MOV     B,A       ;CURRENT DCHARACTER
3449      1331    D6  FD                    SUI     3750      ;COMPARE TO MINUS CHAR
3450      1333    5F                        MOV     E,A       ;CHAR - MINUS SIGN
3451      1334    CA  3D  13                JZ      INP5      ;IF MINUS SIGN
3452      1337    C6  02                    ADI     2         ;COMPARE TO PLUS CHAR
3453      1339    78                        MOV     A,B       ;CURRENT CHARACTER
3454      133A    C2  3F  13                JNZ     INP6      ;IF NOT PLUS SIGN
3455      133D    2C              INP5:     INR     L         ;TO ADDRESS NEXT CHAR
3456      133E    7E                        MOV     A,M       ;NEXT CHARACTER OF STRING
3457      133F    06  00          INP6:     MVI     B,0       ;POSSIBLE DEC EXPONENT
3458      1341    FE  0A                    CPI     120       ;SET CARRY IF CHAR IS DIGIT
3459      1343    D2  5D  13                JNC     INP8      ;IF CHAR IS NOT A DIGIT
3460      1346    47                        MOV     B,A       ;DEC EXP EQUAL DIGIT
3461      1347    2C                        INR     L         ;TO ADDRESS NEXT CHARACTER
3462      1348    7E                        MOV     A,M       ;NEXT CHARACTER OF STRING
3463      1349    FE  0A                    CPI     120       ;SET CHARRY IF CHAR IS DIGIT
3464      134B    D2  56  13                JNC     INP7      ;IF CHAR IS NOT A DIGIT
3465                              ;     FORM COMPLETE DECIMAL EXPONENT.
3466      134E    4F                        MOV     C,A       ;LS DIGIT OF DEC EXPONENT
3467      134F    78                        MOV     A,B       ;MS DIGIT OF DEC EXP
3468      1350    87                        ADD     A         ;2 * MS DIGIT
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

3469    1351  87                        ADD     A         ;4 * MS DIGIT
3470    1352  80                        ADD     B         ;5 * MS DIGIT
3471    1353  87                        ADD     A         ;;10 * MS DIGIT
3472    1354  81                        ADD     C         ;10 * MS DIGIT + LS DIGIT
3473    1355  47                        MOV     B,A       ;DECIMAL EXPONENT
3474    1356  7B              INP7:     MOV     A,E       ;SIGN OF DEC EXPONENT
3475    1357  A7                        ANA     A         ;SET CONTROL BITS
3476    1358  C2  5D  13               JNZ     INP8      ;IF SIGN PLUS
3477    135B  90                        SUB     B         ;COMPLEMENT DEC EXP
3478    135C  47                        MOV     B,A       ;DECIAML EXPONENT
3479    135D  26  00            INP8:     MVI     H,SCRB    ;TO ADDRESS SCRATCH BANK
3480    135F  2E  3F                    MVI     L,TMP3    ;TO ADDRESS INPUT SIGN
3481    1361  4E                        MOV     C,M       ;INPUT SIGN
3482    1362  2E  36                    MVI     L,ACCS    ;TO ADDRESS ACCUM SIGN
3483    1364  71                        MOV     M,C       ;ACCUMULATOR SIGN
3484    1365  78                        MOV     A,B       ;DECIMAL EXPONENT
3485                    ;       CONVERT DECIMAL EXPONENT TO BINARY
3486    1366  2E  3D            INP9:     MVI     L,TMP1    ;TO ADDR DEC EXPONENT
3487    1368  86                        ADD     M         ;ADJUST DECIMAL EXPONENT
3488    1369  CA  D1  0F               JZ      TST       ;IN DEX EXP IS ZERO
3489    136C  77                        MOV     M,A       ;CURRENT DECIMAL EXPONENT
3490    136D  21  64  14                LXI     H,FTEN    ;TO ADDR FLOATING TEN
3491    1370  F2  7B  13               JP      INP10     ;IF MULTIPLY REQUIRED
3492    1373  CD  2C  10                CALL    DIV       ;DIVIDE BY TEN
3493    1376  3E  01                    MVI     A,1       ;TO INCREMENT DEC EXP
3494    1378  C3  66  13                JMP     INP9      ;TO TEST FOR COMPLETEION
3495    137B  CD  04  10        INP10:    CALL    MUL       ;MULTIPLY BY 10
3496    137E  D8                        RC                ;RETURN IF OVERFLOW
3497    137F  3E  FF                    MVI     A,3770    ;TO DECREMENT DEC EXP
3498    1381  C3  66  13                JMP     INP9      ;TO TEST FOR COMPLETION
3499                    ;       OUT SUBROUTINE ENTRY POINT
3500                    ;       SAVE CHARACTER ADDRESS AND ACCUMULATOR
3501    1384  2D                OU:       DCR     L         ;DECREMENT CHARACTER ADDRESS
3502    1385  CD  4C  14                CALL    SVAD      ;SET CHAR ADDR, DIG CNT, DEC EXP
3503    1388  CD  D1  0F                CALL    TST       ;LOAD ACCUM TO REGISTERS
3504    138B  2E  40                    MVI     L,VALE    ;TO ADDR CACCUM SAVE AREA
3505    138D  CD  B6  0F                CALL    STR       ;CALL REG STR SUBROUTINE
3506                    ;       OUTPUT SIGN CHARACTER
3507    1390  CD  59  14                CALL    CHAD      ;CALL CHAR ADDR SBRTN
3508    1393  36  F0                    MVI     M,3600    ;STORE SPACE CHAR
3509    1395  A7                        ANA     A         ;TEST CONTROL BITS
3510    1396  CA  B2  13                JZ      OUT3      ;IF ACCUMULATOR IS ZERO
3511    1399  5F                        MOV     E,A       ;ACCUMULATOR EXPONENT
3512    139A  78                        MOV     A,B       ;ACCUM SIGN AND 1ST FRACTION
3513    139B  A7                        ANA     A         ;SET CONTROL BITS
3514    139C  7B                        MOV     A,E       ;ACCUM EXPONENT
3515    139D  F2  A2  13                JP      OUT1      ;IF ACCUM IS POSITIVE
3516    13A0  36  FD                    MVI     M,3750    ;CHANGE SIGN TO MINUS
3517                    ;SCALE
3518                    ;       SCALE ACCUMULATOR TO .1 = 1. RANGE
3519    13A2  FE  7E            OUT1:     CPI     1760      ;COMPARE TO SMALL EXPONENT
3520    13A4  21  64  14        OUT2:     LXI     H,FTEN    ;TO ADDR FLOATING TEN
3521    13A7  DA  BC  13                JC      OUT4      ;IF EXPOENT TOO SMALL
3522    13AA  FE  81                    CPI     2010      ;COMPARE TO LARGE EXP
3523    13AC  DA  C7  13                JC      OUT5      ;IF EXP SMALL ENOUGH
```

```
STMNT   LOC   OBJECT CODE     SOURCE STATEMENT

3524    13AF  CD 2C 10              CALL    DIV         ;DIVIDE BY TEN
3525    13B2  26 00           OUT3: MVI     H,SCRB      ;TO ADDR SCRATCH BANK
3526    13B4  2E 3E                 MVI     L,TMP2      ;TO ADDR DECIMAL EXPONENT
3527    13B6  5E                    MOV     E,M         ;DECIMAL EXPONENT
3528    13B7  1C                    INR     E           ;INCREMNT DECIMAL EXPONENT
3529    13B8  73                    MOV     M,E         ;DECIMAL EXPONENT
3530    13B9  C3 A4 13              JMP     OUT2        ;TO TEST FOR SCALING COMPLETE
3531    13BC  CD 04 10        OUT4: CALL    MUL         ;MULTIPLY BY TEN
3532    13BF  2E 3E                 MVI     L,TMP2      ;TO ADDR DECIMAL EXPONENT
3533    13C1  5E                    MOV     E,M         ;DECIMAL EXPONENT
3534    13C2  1D                    DCR     E           ;DECREMENT DECIMAL EXPONENT
3535    13C3  73                    MOV     M,E         ;DECIAMAL EXPOENT
3536    13C4  C3 A2 13              JMP     OUT1        ;TO TEST FOR SCALING COMPLETE
3537                          ;     ROUND THE VALUE BY ADDING .00000005
3538    13C7  CD C8 0F        OUT5: CALL    ABS         ;SET ASCCUM POSITIVE
3539    13CA  21 68 14              LXI     H,RND0      ;TO ADDRESS ROUNDER
3540    13CD  CD 4F 10              CALL    AD          ;ADD THE ROUNDER
3541    13D0  FE 81                 CPI     2010        ;CHECK FOR OVERFLOW
3542    13D2  D2 A4 13              JNC     OUT2        ;IF EXP TOO LARGE
3543                          ;     SET DIGIT COUNT
3544    13D5  2E 3E                 MVI     L,TMP2      ;TO ADDR DECIMAL EXPONENT
3545    13D7  7E                    MOV     A,M         ;EDECIMAL EXPONENT
3546    13D8  5F                    MOV     E,A         ;DIGITS BEFOR DEC POINT
3547    13D9  FE 08                 CPI     0100        ;COMPARE TO LARGE EXP
3548    13DB  DA E0 13              JC      OUT6        ;IF ECXPONENT IN RANGE
3549    13DE  1E 01                 MVI     E,1         ;DIGITS BEFORE DEC POINT
3550    13E0  93              OUT6: SUB     E           ;ADJUST DEC EXPONENT
3551    13E1  77                    MOV     M,A         ;DECIAMAL EXPONENT
3552    13E2  3E 07                 MVI     A,7         ;TOTAL NUMBER OF DIGITS
3553    13E4  93                    SUB     E           ;DIGITS AFETER DECIMAL POINT
3554    13E5  2C                    INR     L           ;TO ADDR 2ND DIGIT CNT
3555    13E6  77                    MOV     M,A         ;DIGITS AFTER DECIAML COPOINT
3556    13E7  1D                    DCR     E           ;DECREMENT DIGIT COUNT
3557    13E8  7B                    MOV     A,E         ;DIGITS BEGORE DEC PNT
3558                          ;     OUTPUT SIGNIFICANT DIGITS.
3559    13E9  2E 3D           OUT7: MVI     L,TMP1      ;TO ADDR DIGIT COUNT
3560    13EB  86                    ADD     M           ;ADJUST DIGIT COUNT
3561    13EC  77                    MOV     M,A         ;NEW DIGIT COUNT
3562    13ED  FA 0A 14              JM      OUT8        ;IF COUNT TO OUT
3563    13F0  21 64 14              LXI     H,FTEN      ;TO ADDR FLOATING TEN
3564    13F3  CD 04 10              CALL    MUL         ;MULTIPLY BY TEN
3565    13F6  1E 08                 MVI     E,100       ;TO PLACE DIGIT IN REG A
3566    13F8  CD 8E 12              CALL    FIX         ;CONVERT TO FIXED FORMAT
3567    13FB  CD 59 14              CALL    CHAD        ;CALL CHAR ADDR SBTN
3568    13FE  77                    MOV     M,A         ;OUTPUT DECIMAL DIGIT
3569    13FF  AF                    XRA     A           ;CLEAR CURRENT DIGIT
3570    1400  1E 08                 MVI     E,0100      ;BINARY SCALING FACROR
3571    1402  CD 77 12              CALL    FLT         ;RESTORE VALUE MINUS DIGIT
3572    1405  3E FF                 MVI     A,3770      ;TO ADJUST DIGIT CNT
3573    1407  C3 E9 13              JMP     OUT7        ;LOOP FOR NEXT DIGIT
3574    140A  2E 3F           OUT8: MVI     L,TMP3      ;TO ADDR 2ND DIGIT CNT
3575    140C  7E                    MOV     A,M         ;DIGITS AFTER DECIMAL POINT
3576    140D  36 FF                 MVI     M,3770      ;SET 2ND COUNT NEGATIVE
3577    140F  A7                    ANA     A           ;SET CONTROL BITS
3578    1410  FA 1D 14              JM      OUT9        ;IF 2ND COUNT RAN OUT
```

```
STMNT    LOC     OBJECT CODE      SOURCE STATEMENT

3579    1413     CD 59 14                 CALL    CHAD        ;CALL CHAR ADDR SBRTN
3580    1416     36 FE                    MVI     M,3760      ;STORE DECIMAL POINT
3581    1418     26 00                    MVI     H,SCRB      ;TO ADDR SCRATCH BANK
3582    141A     C3 E9 13                 JMP     OUT7        ;LOOP FOR NEXT DIGIT
3583    141D     2D               OUT9:   DCR     L           ;TO ADDR DECIMAL EXPONENT
3584    141E     A6                       ANA     M           ;DECIMAL EXPONENT
3585    141F     CA 44 14                 JZ      OUT13       ;IF DECIMAL EXPONENT IS ZERO
3586                              ;       OUTPUT DECIMAL EXPONENT
3587    1422     06 FB                    MVI     B,3730      ;PLUS CHARACTER
3588    1424     F2 2C 14                 JP      OUT10       ;IF EXPONENT IS POSITIVE
3589    1427     06 FD                    MVI     B,3750      ;CHANGE SIGN TO MINUS
3590    1429     4F                       MOV     C,A         ;NEGATIVE EXPONENT
3591    142A     AF                       XRA     A           ;ZERO
3592    142B     91                       SUB     C           ;COMPLEMENT EXPONENT
3593    142C     0E FF            OUT10:  MVI     C,3770      ;EMBRYO TENS DIGIT
3594    142E     57               OUT11:  MOV     D,A         ;UNITS DIGIT
3595    142F     0C                       INR     C           ;INCREMENT TENS DIGIT
3596    1430     D6 0A                    SUI     0120        ;REDUCE REMAINDER
3597    1432     D2 2E 14                 JNC     OUT11       ;IF MORE TENS
3598    1435     3E 15                    MVI     A,0250      ;EXPONENT SIGN
3599    1437     CD 59 14         OUT12:  CALL    CHAD        ;CALL CHAR SBRTN
3600    143A     CD B6 0F                 CALL    STR         ;STORE LAST 4 CHARACTER
3601    143D     26 00                    MVI     H,SCRB      ;TO ADDRESS SCRATCH BANK
3602    143F     2E 40                    MVI     L,VALE      ;TO ADDRESS ACCUM SAVE AREA
3603    1441     C3 E6 0F                 JMP     LOD         ;RESTORE ACCUM AND EXIT
3604                              ;       OUTPUT 4 SPACES IF EXPONENT IS ZERO.
3605    1444     3E F0            OUT13:  MVI     A,3600      ;SPACE CHARACTER
3606    1446     47                       MOV     B,A         ;SPACE CHARACTER
3607    1447     4F                       MOV     C,A         ;SPACE CHARACTER
3608    1448     57                       MOV     D,A         ;SPACE CHARACTER
3609    1449     C3 37 14                 JMP     OUT12       ;TO STORE  CHARACTERS
3610                              ;       SUBROUTINE TO SAVE CHARACTER STRING ADDR
3611    144C     7D               SVAD:   MOV     A,L         ;CHARACTER SRRING WORD
3612    144D     44                       MOV     B,H         ;CHARACTER STRING BANK
3613    144E     0E 00                    MVI     C,0         ;INPUT EXP OFR DIGIT CNT
3614    1450     51                       MOV     D,C         ;DEC PNT FLAG OR DFC EXP
3615    1451     26 00                    MVI     H,SCRB      ;TO ADDRESS SCRATCH BANK
3616    1453     2E 3B                    MVI     L,ADRL      ;TO ADDRESS STRING WORD
3617    1455     CD B6 0F                 CALL    STR         ;STORE A, B, C, AND D
3618    1458     C9                       RET                 ;RETURN TO CALLER
3619                              ;       SUBROUTINE TO OBTAIN NEXT CHARACTER ADDR
3620    1459     26 00            CHAD:   MVI     H,SCRB      ;TO ADDRESS SCRATCH BANK
3621    145B     2E 3B                    MVI     L,ADRL      ;TO ADDR CHAR STRING WORD
3622    145D     5E                       MOV     E,M         ;CHARACTER STRING WORD
3623    145E     1C                       INR     E           ;TO ADDR NEXT CHAR
3624    145F     73                       MOV     M,E         ;UPDATE CHAR STRING WORD
3625    1460     2C                       INR     L           ;TO ADDR CHAR STRING BANK
3626    1461     66                       MOV     H,M         ;CHARACTER STRING BANK
3627    1462     6B                       MOV     L,E         ;CHARACTER STRING WORD
3628    1463     C9                       RET                 ;RETURN TO CALLER
3629    1464              84     FTEN:   DB      2040        ;FLOATING TEN
3630    1465              20             DB      0400
3631    1466              00             DB      0
3632    1467              00             DB      0
3633    1468              68     RND0:   DB      1500        ;.00000005
```

```
STMT    LOC     OBJECT CODE     SOURCE STATEMENT

3634    1469            56              DB      1260
3635    146A            BF              DB      2770
3636    146B            AD              DB      2550
3637    146C            00              DB      0           ;CHECKSUM WORD
3638
3639                                    ;CONSTANTS VARIABLES AND TABLES
3640    0004                    WORDS   EQU     4           ;LENGTH
3641    007F                    RUBOUT: EQU     7FH
3642    0002                    TEST:   EQU     2           ;UART STATUS PORT
3643    0003                    CRT:    EQU     3           ;CRT OUTPUT PORT
3644    0020                    SP:     EQU     ' '
3645    000D                    CR:     EQU     150         ;CARRIAGE RETURN
3646    000A                    LF:     EQU     120
3647    0000                    KYBRD   EQU     0           ;KEYBOARD OUTPUT UH, INPUT OPORT
3648    146D            00  00  AXIN:   DW      0           ;STORAGE INDEX
3649    146F                    TEXTP:  EQU     $
3650    146F            29  16  AXOUT:  DW      FRSTX       ;OUTPUT INDEX
3651    1471            64  15  PC:     DW      FLTZER      ;PROGRAM COUNTER
3652    1473            00  00  THISLN: DW      0           ;LINE POINTER FROM FINDLN
3653    1475            00  00  LASTLN: DW      0           ;BACK POINTER FROMA FINDL
3654    1477            00      DEBGSW: DB      0           ;DEBUG SWITCH; NONXZERO FOR LITERAL
3655    1478            00  00  PT1:    DW      0           ;VARIABLE POINTER
3656    147A            2A  16  LASTV:  DW      BUFBEG      ;ADDRESS OF LASTE VARIABLE
3657    147C            00      SRTCN:  DB      0           ;RESULT OF SORTDC
3658    147D            00      LASTOP: DB      0           ;LAST OPERATION FOR EVAL
3659    147E                    EFOP:   EQU     $           ;FUNCTION CODE
3660    147E            00      ATSW:   DB      0           ;ASK-TYPE SWITCH
3661    147F            F0      CNTR:   DB      -200        ;DELETE AND ERROR COUNTER
3662    1480            FF              DB      0FFH
3663    1481            00      STE:    DB      0
3664    1482                    STRTV:  EQU     $
3665    1482            2A  16  BUFR:   DW      BUFBEG      ;NEXT LOC
3666    1484            01      NAGSW:  DB      1           ;SWITCH (200=ONE, 1=ALL, 0=GROUP)
3667    1485            0D      CHAR:   DB      150         ;CHARACTER SAVE
3668    1486            00  00  LINENO: DW      0           ;LINE NO FROM GETLN (BCD)
3669    1488                    LIST6:  EQU     $           ;INPUT LIST FOR SFOUND
3670    1488            0C              DB      140         ;F.F.
3671    1489            07              DB      7           ;BELL
3672    148A                    LIST7:  EQU     $
3673    148A            03              DB      3           ;CONTROL C FOR DEBUGGING
3674    148B            5F              DB      1370        ;LEFT ARROW
3675    148C            0A              DB      120         ;LINE FEED
3676    148D                    LIST3:  EQU     $           ;EXCRETION LIST
3677    148D            0D              DB      150         ;LIST BRANCHER
3678    148E            00              DB      0
3679    148F            00              DB      0           ;END OF LIST
3680    1490                    OPTBL:  EQU     $           ;FLOATING POINT CALL ADDRESSES
3681    1490            E3  03          DW      RECOVER
3682    1492            4F  10          DW      AD
3683    1494            4C  10          DW      SB
3684    1496            2C  10          DW      DIV
3685    1498            04  10          DW      MUL
3686    149A            60  0F          DW      FLEX
3687    149C            E3  03          DW      RECOVER
3688    149E            00      DMPSW:  DB      0           ;SEARCH CHAR-VARIABLE; 0-TRACE
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

3689    149F                    FNTBF:  EQU     $       ;FUNCTION ADDRESSES
3690    149F            0D 0F           DW      XABS    ;ABSOLUTE VALUE
3691    14A1            34 0F           DW      XSGN    ;SIGN PART
3692    14A3            1D 0F           DW      XINT    ;INTEGER PART
3693    14A5            E0 0E           DW      XRAN    ;RANDOM NUMBER
3694    14A7            AB 0C           DW      ARTN    ;ARCTANGENT
3695    14A9            D7 0D           DW      FEXP    ;EXPONENTIAL FUNCTIONS
3696    14AB            1A 0E           DW      FLOG    ;
3697    14AD            11 0C           DW      FSIN    ;TRIG FUNCTIONS
3698    14AF            FF 0B           DW      FCOS
3699    14B1            98 0E           DW      XSQT    ;SQUARE ROOT
3700    14B3            72 0B           DW      XUSR
3701    14B5            3F 0D           DW      FHYS    ;HYPERBOLIC SIN
3702    14B7                    FNTBL:  EQU     $       ;LIST OF CODED FUNCTION NAMES
3703    14B7                42          DB      'B'     ;ABS
3704    14B8                47          DB      'G'     ;SGN
3705    14B9                4E          DB      'N'     ;ITR
3706    14BA                41          DB      'A'     ;RAN
3707    14BB                54          DB      'T'     ;ARCTAN
3708    14BC                58          DB      'X'     ;EXP
3709    14BD                4C          DB      'L'     ;LOG
3710    14BE                49          DB      'I'     ;SIN
3711    14BF                4F          DB      'O'     ;COS
3712    14C0                51          DB      'Q'     ;SQT
3713    14C1                53          DB      'S'     ;USER
3714    14C2                59          DB      'Y'     ;HYPERBOLIC SIN
3715    14C3                00          DB      0       ;END
3716    14C4                    INLIST: EQU     $       ;INPUT CONTROL CHARACTERS
3717    14C4            E3 03           DW      RECOVER ;C.C=BREAK
3718    14C6            84 04           DW      IBAR    ;B.A.=RESTART
3719    14C8            93 04           DW      IGNOR   ;L.F.=IGNORE
3720    14CA            A6 04           DW      IRETN   ;C.R.=TERMINATE STRING
3721    14CC            FC 06   FLST2:  DW      FLIMIT  ;=STANDARD
3722    14CE            83 07           DW      FINFIN  ;=SHORT
3723    14D0            DC 03           DW      ERROR5  ;CR=DUMB
3724    14D2            E5 06   FLST1:  DW      FINCR   ;=STANDARD FORMAT
3725    14D4            B4 05           DW      PROCESS ;=SET; PLUS,...
3726    14D6            BF 05           DW      PC1     ;C.R.=SET COMMAND
3727    05BF                    COMMENTS EQU    PC1     ;ALSO CONTINUE
3728    14D8            A6 06   ILIST:  DW      IF1     ;,
3729    14DA            B4 05           DW      PROCESS ;;
3730    14DC            BF 05           DW      PC1     ;CR
3731    14DE                    COMLST: EQU     $       ;COMMAND DECODE LIST-SPEED OPTIMIZED
3732    14DE                53          DB      'S'     ;SET
3733    14DF                46          DB      'F'     ;FOR
3734    14E0                49          DB      'I'     ;IF
3735    14E1                44          DB      'D'     ;DO
3736    14E2                47          DB      'G'     ;GO TO
3737    14E3                43          DB      'C'     ;COMMENT
3738    14E4                41          DB      'A'     ;ASK
3739    14E5                54          DB      'T'     ;TYPE
3740    14E6                4C          DB      'L'     ;LIBRARY
3741    14E7                45          DB      'E'     ;ERASE
3742    14E8                57          DB      'W'     ;WRITE
3743    14E9                4D          DB      'M'     ;MODIFY
```

```
STMNT   LOC   OBJECT CODE      SOURCE STATEMENT

3744    14EA            51              DB      'Q'         ;QUIT
3745    14EB            52              DB      'R'         ;RETURN
3746    14EC            4A              DB      'J'         ;JUMP
3747    14ED            00              DB      0           ;END
3748    14EE                    COMGO:  EQU     $           ;COMMAND ROUTINE ADDRESSES
3749    14EE        B3  06              DW      SET
3750    14F0        B3  06              DW      FOR
3751    14F2        76  06              DW      IF
3752    14F4        20  05              DW      DO
3753    14F6        A5  05              DW      GOTO
3754    14F8        BF  05              DW      COMMENT
3755    14FA        95  07              DW      ASK
3756    14FC        97  07              DW      TYPE
3757    14FE        B3  0F              DW      LIBRARY
3758    1500        FE  0A              DW      ERASE
3759    1502        F8  05              DW      WRITE
3760    1504        54  08              DW      MODIFY
3761    1506        64  04              DW      START       ;QUIT
3762    1508        6A  00              DW      RETRN
3763    150A        67  06              DW      JUMP        ;COMPUTED TRANSFER
3764    150C                    SRNLST: EQU     $           ;MODIFY CONTROL TABLE
3765    150C        7D  08              DW      SCHAR       ;F.F.=CONTINUE
3766    150E        72  08              DW      SCONT       ;BELL=CHANGE SEARCH CHAR
3767    1510        E3  03              DW      RECOVER     ;C.C.=BREAK
3768    1512        94  08              DW      SBAR        ;B.A.=RESTART
3769    1514        75  08              DW      SCONT+3     ;L.F.=FINISH LINE AS WAS
3770    1516                    LISTGO: EQU     $
3771    1516        F0  04              DW      SRETN       ;C.R.=END LINE HERE, AS IS
3772    1518        AB  08              DW      SGOT        ;CHAR=SEARCH CHAR
3773    151A                    ALIST:  EQU     $           ;ASK-TYPE LIST OF CONTROLS
3774    151A            25              DB      '%'
3775    151B            22              DB      '"'
3776    151C            21              DB      21H         ;)
3777    151D            23              DB      430
3778    151E            24              DB      '$'
3779    151F            20      GLIST:  DB      ' '
3780    1520            2C      TLIST:  DB      ','
3781    1521            3B              DB      ';'
3782    1522            0D              DB      CR
3783    1523            00              DB      0
3784    1524        34  12      RAN0:   DW      1234H
3785    1526        34  12              DW      1234H
3786    1528                    ATLIST: EQU     $           ;ASK-TYPE CONTROL CHAR TABLE
3787    1528        35  08              DW      TINTR       ;%-FORMAT DELIMITER
3788    152A        1C  08              DW      TQUOT       ;"-LITERAL DELIMITER
3789    152C        44  08              DW      TCRL2       ;)-CR&LF
3790    152E        49  08              DW      TCRLF       ;#-CR ONLY
3791    1530        67  08              DW      TDUMP       ;$-DUMP SYMBOL TABLE
3792    1532        4E  08              DW      TASK4       ;SP-TERMINATOR FOR NAMES
3793    1534        4E  08              DW      TASK4       ;,-TERMINATOR FOR EXPRESSIONS
3794    1536        B4  05              DW      PROCESS     ;;-TERMINATOR FOR COMMAND
3795    1538        BF  05              DW      PC1         ;CR-TERMINATOR FOR STRINGS
3796    153A            0D      ECHOLST: DB     CR
3797    153B            7F              DB      RUBOUT
3798    153C            00              DB      0
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

3799     153D                    TERMS:  EQU     $       ;TERMINATOR TABLE
3800     153D           20               DB      SP
3801     153E           2B               DB      '+'
3802     153F           2D               DB      '-'
3803     1540           2F               DB      '/'
3804     1541           2A               DB      '*'
3805     1542           5E               DB      5EH     ;UP ARR
3806     1543           28               DB      '('
3807     1544           5B               DB      5BH
3808     1545           3C               DB      74O
3809     1546           29               DB      ')'
3810     1547           5D               DB      5DH
3811     1548           3E               DB      76O
3812     1549           2C               DB      ','
3813     154A           3B               DB      ';'
3814     154B           0D               DB      CR
3815     154C           3D               DB      '='     ;TO END GFRTARG FROM SET
3816     154D           00               DB      0       ;END LIST
3817     154E     00    00       FLARG:  DW      0       ;DATA STORAGE
3818     1550     00    00               DW      0
3819     1552                    TLST3:  EQU     $       ;LITERAL TERMINATORS
3820     1552     4E    08               DW      TASK4   ;"
3821     1554     BF    05               DW      PC1     ;CR=AUTOMATINC QUOTE MATCH
3822     1556                    INFIX:  EQU     $       ;DATA CONTROL CHARACTERS
3823     1556     DC    03               DW      ERROR4  ;LEFT ARROW=KILL
3824     1558     C4    07               DW      AK2     ;RUBOUT=IGNORE
3825     155A     C4    07               DW      AK2     ;LF=IGONORE
3826     155C     F9    07               DW      AK4     ;ALTNODE
3827     155E     D7    07               DW      AK3     ;CR INFIN
3828     1560     E3    03               DW      RECOVER ;C.C.
3829     1562           81       FLTONE: DB      81H
3830     1563           00               DB      0
3831     1564     00    00       FLTZER: DW      0
3832     1566     00    00               DW      0
3833     1568           82       FLTTWO: DB      82H
3834     1569           00               DB      0
3835     156A     00    00               DW      0
3836     156C                    FLAC:   DS      4
3837     1570           44       FISW:   DB      1040    ;FLOATING OUTPUT FORMAT
3838     1571                    SPECIAL: EQU    $       ;INPUT CHARACTERS
3839     1571           5F               DB      137O    LEFT ARROW
3840     1572           7F               DB      RUBOUT
3841     1573           0A               DB      LF
3842     1574           1B               DB      33O     ;ALT MODE
3843     1575           0D               DB      CR      ;C.R.
3844     1576           03               DB      3       ;C.C.
3845     1577           00               DB      0       END
3846     1578     00    00       ITER1:  DW      0
3847     157A     00    00       TEMP:   DW      0       ;TEMP STORAGE
3848     157C     00    00       TEMP2   DW      0
3849     157E           22       TLST2:  DB      '"'     ;QUOTE
3850     157F           0D               DB      CR      ;FUNCTION OR NUM8IS NOT AN ARG
3851     1580           00               DB      0
3852     1581           FF               DB      0FFH
3853     1582           20       DATBUF: DC      '
```

```
STMNT     LOC     OBJECT CODE      SOURCE STATEMENT.
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
3854      15AA            FF                  DB      3770      ;OUTCHAR. BUFFER
3855      15AB            20       IOBUF:     DC      '
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
```

```
STMNT    LOC    OBJECT CODE      SOURCE STATEMENT

                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
3856     15D3            FF      COMEIN:  DB     377Q
3857     15D4            20               DC '
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
                         20
```

```
STMNT     LOC     OBJECT CODE      SOURCE STATEMENT
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
3858      1606            20                   DC      '                    '
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
                          20
3859      1624                       COMEOUT:  EQU     $
3860      1624            FF                   DB      3770
3861      1625        00  00         FRST:     DW      0          ;TEXT POINTER
3862      1627        00  00                   DW      0          ;DUMMY LINE NUMBER
```

```
STMNT    LOC     OBJECT CODE      SOURCE STATEMENT

3863     1629          0D        FRSTX:   DB      CR          ;DUMMY END OF LINE
3864     162A                    BUFBEG:  EQU     $
3865     15D4                    COMBUF:  EQU     COMEIN+1    ;COMMAND BUFFER START
3866     1625                    CFRS:    EQU     FRST        ;ADDRESS OF FISRT LINE (DUMMY)
3867     1564                    CFRSX:   EQU     FLTZER      ;POINTER TO ZERO DATA
3868     15AB                    OPTR0:   EQU     IOBUF       ;OUTPUT POINTERS
3869     0005                    SCR:     EQU     SBANK
3870     0000                    SCRB:    EQU     <SBANK      ;BANK NO. OF SCRATCH PAD
3871     0045                    FSQRN:   EQU     SCR+40H
3872     0049                    FSQRX:   EQU     SCR+44H
3873     0045                    IDVT     EQU     SCR+40H
3874     0045                    FMACX    EQU     SCR+40H
3875     0049                    FMACS    EQU     SCR+44H
3876     004D                    FMACT    EQU     SCR+48H
3877     0051                    FMACG    EQU     SCR+4CH
3878     0055                    FSINX:   EQU     SCR+50H
3879     0055                    FATNT:   EQU     SCR+50H
3880     0059                    FATNU:   EQU     SCR+54H
3881     0053                    FSNHD    EQU     SCR+4EH
3882     0054                    FEXOV    EQU     SCR+4FH
3883     0055                    FSNHX    EQU     SCR+50H
3884     0053                    FLOGE:   EQU     SCR+4EH
3885     0050                    FLOGX    EQU     SCR+50H
3886                              END
```

ABS	0FC8	ACC1	0037	ACC2	0038	ACC3	0039	ACCE	0035
ACCS	0036	AD	104F	ADD10	10E3	ADD11	10EC	ADD12	10F2
ADC17	10FE	ADD2	1086	ADD3	10A5	ADD9	10D5	ADRH	003C
ADRL	0038	AK2	07C4	AK3	07D7	AK4	07F9	AK5	07F2
ALIST	151A	ALL	02A5	ARITH	0FB4	ARTHB	000F	ARTN	0CAB
ASK	0795	ATLIS	1528	ATSW	147E	AXIN	146D	AXOUT	146F
B1	02D1	B3	02D0	BUFBE	162A	BUFR	1482	CFRS	1625
CFRSX	1564	CHAD	1459	CHAR	1485	CHS	0FC5	CNTR	147F
COMBU	15D4	COMEI	15D3	COMEO	1624	COMGO	14EE	COMLS	14DE
COMME	05BF	COMP	1167	COMP1	116C	CR	000D	CRT	0003
D1	033A	D2	0398	D3	0382	D4	0386	DATBU	1582
DCCNT	059A	DEBGS	1477	DELET	0329	DGRP	052B	DGRP1	054F
DIV	102C	DIVX	1208	DIVX1	1253	DIVX2	1257	DIVX3	126A
DIVX4	126E	DIVX5	0012	DIVX6	0023	DMPSW	149E	DN1	03CF
DO	0520	DOK	036C	DONE	039F	DOONE	0588	DR2	057E
DR3	0575	E3	01FD	ECALL	0980	ECHOL	153A	EDO	0A98
EFAD	0ABC	EFOP	147E	EFST	0AAD	EFUN	0A53	EFUN3	0A8F
ELP	0AB3	ELPAR	0AD9	ELPR2	0AEC	ENDLN	01E0	ENL2	0A28
ENL3	0A11	ENLP	0A08	ENUM	0A04	ER1	0B4B	ERASE	0AFE
ERC	0B37	ERL	0B2E	ERR2	03DC	ERROR	03DC	ERT	0B17
ERV	0B25	ERVX	0B60	ETERM	0A98	ETR2	09FE	ETRM1	09FD
ETRMN	09F5	EVAL	09B2	EVALC	09C3	F2	01A4	F3	01B4
F4	01BF	FATAN	0CB1	FATN1	0CED	FATNT	0055	FATNU	0059
FCCNT	071A	FCOS	0BFF	FCOS0	0C05	FEND3	01C4	FEXOV	0054
FEXP	0DD7	FEXP0	0DDD	FEXPP	0E09	FHYS	0D3F	FINCR	06F5
FINDL	019E	FINDN	01CF	FINFI	0783	FISW	1570	FIX	128E
FIX1	12BC	FLAC	156C	FLARG	154E	FLEX	0F60	FLIMI	06FC
FLLP	0F93	FLN2	0BFB	FLOG	0E1A	FLOG0	0E20	FLOGE	0053
FLOGX	0050	FLOP	0AC6	FLST1	14D2	FLST2	14CC	FLT	1277
FLTON	1562	FLTTW	1568	FLTZE	1564	FMACA	0BBF	FMACB	0BC8
FMACC	0B9F	FMACD	0BA4	FMACE	0B90	FMACF	0BDE	FMACG	0051
FMACL	0B85	FMACS	0049	FMACT	004D	FMACX	0045	FNDVR	0938
FNTBF	149F	FNTBL	14B7	FONE	0BF3	FOR	06B3	FORMR	0776
FRIV2	0BF7	FRST	1625	FRSTX	1629	FSIN	0C11	FSIN0	0C17
FSINA	0C3D	FSINB	0C71	FSINC	0CA1	FSINH	0D45	FSINX	0055
FSNHA	0D63	FSNHB	0D98	FSNHC	0DC9	FSNHD	0053	FSNHX	0055
FSQRE	0ED2	FSQRL	0EB3	FSQRN	0045	FSQRT	0E9E	FSQRX	0049
FTEN	1464	FUN2	0A75	FUN3	0A63	FUN4	0A7B	GERR	02B6
GETC	008D	GETC2	0090	GETLN	0211	GETVA	08B1	GLIST	151F
GONE	04B0	GOTO	05A5	GROUP	02A0	GT1	0247	GT2	025A
GT3	0281	GT4	0292	GV2	08D7	GV2A	08E9	GV2B	08DA
GV3	093B	GV4	0942	GV5	0917	GV6	092C	GV6A	0920
GZERR	028F	HC	0965	I3	0693	IBAR	0484	IDV	0873
IDVT	0045	IF	0676	IF1	06A6	IF3	068D	IGNOR	0493
ILIST	14D8	IN8	0436	INFIX	1556	INIT	0FB4	INLIS	14C4
INP	12C2	INP1	12DF	INP10	137B	INP2	12E5	INP3	1322
INP4	132C	INP5	133D	INP6	133F	INP7	1356	INP8	135D
INP9	1366	INPTX	0504	INTEL	0000	INTRP	0000	IOBUF	15AB
IP	0691	IRETN	04A6	ITER1	1578	I2	0692	JUMP	0667
KYBRD	0000	LASTL	1475	LASTO	147D	LASTV	147A	LF	000A
LIBRA	0FB3	LINEN	1486	LIST3	148D	LIST6	1488	LIST7	148A
LISTG	1516	LOD	0FE6	LOD1	0FF3	LSH	1134	LSH1	1137
MATCH	00F8	MDEX	110D	MODIF	0854	MUL	1004	MULP1	000E
MULP2	000A	MULP3	0006	MULX	11C5	MULX1	11DD	MULX2	11F0
MULX3	11F1	MULX4	0005	MULX5	11EB	NAGSW	1484	NOPE	095E

NOPE1	095B	NOPE2	095C	NOPE3	095D	NORM	117A	NORM1	117C
NORM2	118D	NORM3	119A	NOSUB	0932	NOTA	08D1	NUMD	0A2F
OLPTS	076F	OP1A	002C	OP1S	001E	OP2A	002B	OP2S	001A
OP3A	0024	OP3S	0016	OP4A	0021	OP4S	0013	OP4X	002F
OPNEX	09D8	OPTBL	1490	OPTR0	15AB	OU	1384	OUT1	13A2
OUT10	142C	OUT11	142E	OUT12	1437	OUT13	1444	OUT2	13A4
OUT3	13B2	OUT4	13BC	OUT5	13C7	OUT6	13E0	OUT7	13F9
OUT8	140A	OUT9	141D	OVER	0033	OVERF	1042	OVUN	1130
P4	05CD	PACKC	00BD	PC	1471	PC1	05BF	PC2	05C0
PC*	05DC	PC3A	05EB	PCHK	0441	PCHK2	0461	PF1	0074
POPF	007F	POPF1	0084	PREX	0034	PRNT	017B	PRNTC	011B
PRNTL	0164	PROC	05B7	PROCE	05B4	PROCT	0001	PT1	147E
PTC1	0122	PTC2	0139	PUSHF	0071	PUSHJ	0066	R2	03F1
R3	03E9	R4	0426	RAN0	1524	READC	0151	RECOV	03F3
RETRN	006A	RN1	0FE6	RN2	0EF1	RND0	1468	RNDA	1021
RNFR	11B7	ROND	11A8	RSH	1141	RSH0	1143	RSH1	1145
RSH2	1152	RSH3	1155	RUB1	00D5	RUBOU	007F	SB	104C
SBANK	0005	SBAR	0894	SCHAR	087D	SCONT	0872	SCR	0005
SCRB	0000	SET	06B3	SEXC	0116	SF	003A	SFOUN	089E
SGCT	08AB	SORTC	00FE	SORTJ	00E3	SP	0020	SPAD	0003
SPECI	1571	SPNOR	0205	SRC1	0111	SRC2	0103	SRETN	04F0
SRALS	150C	SRT1	00EB	SRT2	00EA	SRTCN	147C	START	0464
STAT	042E	STE	1481	STR	0FB6	STR0	0FB4	STR1	0FB7
STRTV	1482	SVAD	144C	T2	02E5	TASK	079A	TASK4	084E
TBF	0146	TCRL2	0844	TCRLF	0849	TDUMP	0B67	TEMP	157A
TEMP2	157C	TERMS	153D	TEST	0002	TESTC	02F4	TESTN	02B9
TEXTP	146F	THISL	1473	TINTR	0835	TLIST	1520	TLST2	157E
TLST3	1552	TMP1	003D	TMP2	003E	TMP3	003F	TMP4	0044
TOS	0821	TQUOT	081C	TST	0FD1	TST1	0FD3	TSTGR	02F7
TSTLP	02D3	TYP	080E	TYPE	0797	TYPE2	07FD	UT2	00B2
UT*	00AE	VAL1	0041	VAL2	0042	VAL3	0043	VALE	0040
VARGE	09CF	W3	0611	W4	0638	W5	0600	W6	0652
WALL	063E	WORDS	0004	WRITE	05F8	WTEST	064A	WTST2	0629
WX	0656	XABS	0F0D	XI33	0436	XINT	0F1D	XOUTL	0145
XRAN	0EE0	XSG1	0F50	XSG2	0F54	XSG3	0F49	XSGN	0F34
XSGT	0E98	XT2	031F	XT3	0325	XUSR	0B72	XXX	0000
ZFFX	0FA6	ZR0	0FBE	ZR01	0FC0				

0 ERROR(S) DETECTED DURING PASS 1
0 ERROR(S) DETECTED DURING PASS 2